



DOCTORAL THESIS

Fuzzy Model Predictive Control. Complexity Reduction by Functional Principal Component Analysis

Author:

Juan Manuel Escaño
González

Supervisor:

Dr. Carlos Bordons

*A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy*

in the

Departamento de Ingeniería de Sistemas y Automática

July 2015

Abstract

Fuzzy Model Predictive Control. Complexity Reduction by Functional Principal Component Analysis

Juan Manuel Escaño González

In Model-based Predictive Control, the controller runs a real-time optimisation to obtain the best solution for the control action. An optimisation problem is solved to identify the best control action that minimises a cost function related to the process predictions. Due to the computational load of the algorithms, predictive control subject to restrictions is not suitable to run on any hardware platform. Predictive control techniques have been well known in the process industry for decades. The application of advanced control techniques based on models is becoming increasingly attractive in other fields such as building automation, smart phones, wireless sensor networks, etc., as the hardware platforms have never been known to have high computing power.

The main purpose of this thesis is to establish a methodology to reduce the computational complexity of applying nonlinear model based predictive control systems subject to constraints, using as a platform hardware systems with low computational power, allowing a realistic implementation based on industry standards. The methodology is based on applying the *functional principal component analysis*, providing a mathematically elegant approach to reduce the complexity of rule-based systems, like fuzzy and piece wise affine systems, allowing the reduction of the computational load on model-based predictive control systems, subject or not subject to constraints.

The idea of using fuzzy inference systems, in addition to allowing nonlinear or complex systems modelling, endows a formal structure which enables implementation of the aforementioned complexity reduction technique.

This thesis, in addition to theoretical contributions, describes the work done with real plants on which tasks of modeling and fuzzy control have been carried out. One of the objectives to be covered for the period of research and development of the thesis has been training with fuzzy systems and their simplification and application to industrial systems. The thesis provides a practical knowledge framework, based on experience.

Resumen de la tesis

Control Predictivo basado en Modelos Borrosos. Reducción de la complejidad mediante el Análisis de Componentes Principales Funcionales

Juan Manuel Escaño González

En el Control Predictivo basado en Modelo, el controlador ejecuta una optimización en tiempo real para obtener la mejor solución para la acción de control. Un problema de optimización se resuelve para identificar la mejor acción de control que minimiza una función de coste relacionada con las predicciones de proceso. Debido a la carga computacional de los algoritmos, el control predictivo sujeto a restricciones, no es adecuado para funcionar en cualquier plataforma de hardware. Las técnicas de control predictivo son bien conocidos en la industria de proceso durante décadas. Es cada vez más atractiva la aplicación de técnicas de control avanzadas basadas en modelos a otros muchos campos tales como la automatización de edificios, los teléfonos inteligentes, redes de sensores inalámbricos, etc., donde las plataformas de hardware nunca se han conocido por tener una elevada potencia de cálculo.

El objetivo principal de esta tesis es establecer una metodología para reducir la complejidad de cálculo al aplicar control predictivo basado en modelos no lineales sujetos a restricciones, utilizando como plataforma, sistemas de hardware de baja potencia de cálculo, permitiendo una implementación basado en estándares de la industria.

La metodología se basa en la aplicación del análisis de componentes principales funcionales, proporcionando un enfoque matemáticamente elegante para reducir la complejidad de los sistemas basados en reglas, como los sistemas borrosos y los sistemas lineales a trozos. Lo que permite reducir la carga computacional en el control predictivo basado en modelos, sujetos o no a restricciones.

La idea de utilizar sistemas de inferencia borrosos, además de permitir el modelado de sistemas no lineales o complejos, dota de una estructura formal que permite la implementación de la técnica de reducción de la complejidad mencionada anteriormente.

En esta tesis, además de las contribuciones teóricas, se describe el trabajo realizado con plantas reales en los que se han llevado a cabo tareas de modelado y control borroso. Uno de los objetivos a cubrir en el período de la investigación y el desarrollo de la tesis ha sido la experimentación con sistemas borrosos, su simplificación y aplicación a sistemas industriales. La tesis proporciona un marco de conocimiento práctico, basado en la experiencia.

Acknowledgements

Firstly I would like to thank my supervisor Dr. Carlos Bordons for his dedication, support, patience and confidence to do this work. Also I would like to thank my family, especially my mother and my brother Dr. Carlos Escaño who have been supporting this work (and everyone else) from the beginning until the last moment.

For so many years combining this work with my working life, I have received support and help from many people who in one way or another, have left their mark on me. These include Antonio Nuevo, my friend and control and automation teacher, always available to help in good and bad times, Dr. Eduardo F. Camacho, whose support and help through my research career has been invaluable, Dr. Martin J. Hayes, who welcomed and looked after me, selflessly, in his University, my friend and teacher Dr. Carmine Luca Iandoli from whom I have learned the industrial reality of advanced control and my friend and brother Dr. Kritchai Withephanich, with whom I have shared my research and he has been a support for me through good and bad moments. I especially want to thank my friends Dr. Niel Canty, Dr. Samira Roshany Yamchi, Jean-Michel Rubillon and Dr. Mairtín O'Droma for their extensive revisions, unconditional help, advice and dedication.

During my experience as a lecturer, I have dealt with many students, all of whom I have learned something from, but I must especially thank the future PhD Adolfo Juan Sánchez del Pozo Fernández for his invaluable support during this thesis. I also appreciate the support and time of my dear friends Maria Robledo, Elena González, Samara González and Cristina Martín. I would also like to thank my colleagues and friends in the Department of Systems and Control Engineering of the University of Seville especially Dr. Miguel Ángel Ridao, Dr. Fernando Dorado, Dr. David Muñoz de la Peña, Dr. José María Maestre, Dr. Ignacio Alvarado, Dr. Daniel Limón, Dr. Manuel Gil Ortega, Dr. Fernando Castaño, Dra. Amparo Núñez, Dr. Teodoro Álamo and many others for their advice and contributions to this thesis. Also I also thank Aóife Moloney and Louis Cronin for their assistance with corrections.

I would finally like to thank Richard Linger and Dr. Dirk Pesch for the time that they have given me to finish this thesis at the most critical moment.

Agradecimientos

Primeramente me gustaría agradecer a mi director de tesis Dr. Carlos Bordons su dedicación, soporte, paciencia y confianza para poder realizar este trabajo. También en primer lugar de los agradecimientos está mi familia, especialmente mi madre y mi hermano Dr. Carlos Escaño quienes han estado apoyando este trabajo (y todos) desde el principio y hasta el último instante.

Durante tantos años compaginando este trabajo con mi vida laboral, he recibido el apoyo y ayuda de muchas personas que de una manera o de otra, han dejado huella en mí. Entre ellas están Antonio Nuevo, mi amigo y maestro en control y automatización, siempre disponible a ayudar en buenos y malos momentos, El Dr. Eduardo F. Camacho, cuyo soporte y ayuda en mi carrera investigadora ha sido inestimable, Dr. Martin J. Hayes, que me acogió en su universidad desinteresadamente, mi amigo y maestro Dr. Carmine Luca Iandoli del cual he aprendido la realidad industrial del control avanzado y mi amigo y hermano Dr. Kritchai Withephanich, con el que he compartido mi investigación con buenos y malos momentos y siempre ha sido un soporte para mí.

Quiero agradecer especialmente a mis amigos Dr. Niel Canty, a la Dra. Samira Roshany-Yamchi, Jean Michel Rubillon y Dr. Mairtín O'Droma por sus extensas revisiones, su incondicional ayuda, consejo y dedicación.

Durante mi experiencia como profesor, he tratado con muchísimos alumnos, de todos he aprendido algo, pero debo de agradecer especialmente a mi amigo el futuro doctor Adolfo Juan Sánchez del Pozo Fernández su ayuda inestimable para esta tesis. También agradezco su apoyo y tiempo a a mis queridas María Robledo, Elena González, Samara González y Cristina Martín.

Quisiera también agradecer a mis colegas y amigos del departamento de Ingeniería de Sistemas y Automática de la Universidad de Sevilla especialmente al Dr. Miguel Ángel Ridaó, Dr. Fernando Dorado, Dr. David Muñoz de la Peña, Dr. José María Maestre, Dr. Ignacio Alvarado, Dr. Daniel Limón, Dr. Manuel Gil Ortega, Dr. Fernando Castaño, Dra. Amparo Núñez, Dr. Teodoro Álamo y tantos otros, por sus consejos y contribuciones para esta tesis. También agradezco a Aóife Moloney y Louis Cronin, por sus correcciones.

Quisiera agradecer finalmente a Richard Linger y al Dr. Dirk Pesch el tiempo que me han concedido para terminar la tesis en el momento más crítico.

Contents

| | |
|---|-------------|
| Abstract | i |
| Acknowledgements | iii |
| Contents | v |
| Acronyms | viii |
| | |
| Introduction | 1 |
| 1 Model predictive control and its implementation | 6 |
| 1.1 Model-based predictive control | 6 |
| 1.2 Industrial implementation of MPC | 10 |
| 1.3 MPC for low-cost hardware: a practical implementation perspective | 11 |
| 1.3.1 Ambulatory sensor network description | 12 |
| 1.3.2 Problem formulation | 15 |
| 1.3.3 MPC | 18 |
| 1.3.4 Robust MPC | 19 |
| 1.3.5 Experimental results | 21 |
| 1.4 Conclusion to the chapter | 24 |
| 2 Implementation of fuzzy inference systems | 25 |
| 2.1 Fuzzy logic | 25 |
| 2.1.1 Mamdani fuzzy systems | 27 |
| 2.1.2 Takagi-Sugeno fuzzy systems | 29 |
| 2.2 Industrial standardisation | 30 |
| 2.3 Development of a FIS application for authentication gestures | 32 |
| 2.3.1 Result using the FME | 34 |
| 2.4 Conclusion to the chapter | 38 |
| 3 Fuzzy modelling techniques and applications | 39 |
| 3.1 Fuzzy modelling | 40 |

| | | |
|----------|--|------------|
| 3.1.1 | Takagi-Sugeno models | 42 |
| 3.1.2 | Input selection | 44 |
| 3.1.3 | Fuzzy Time Series | 46 |
| 3.2 | Review of methods to build and train fuzzy systems | 48 |
| 3.2.1 | Clustering methods | 48 |
| 3.2.2 | Back propagation algorithms | 49 |
| 3.2.3 | Evolutionary algorithms | 51 |
| 3.3 | Development and validation of fuzzy models in real applications | 53 |
| 3.3.1 | Autoclave for food sterilization | 54 |
| 3.3.2 | Gas Mixing Chamber | 56 |
| 3.4 | Conclusion of the chapter | 61 |
| 4 | Fuzzy control systems: practical implementation | 63 |
| 4.1 | Direct fuzzy controllers | 64 |
| 4.1.1 | Illustrative application: pneumatic levitation system | 68 |
| 4.1.2 | Stability analysis of fuzzy control systems | 72 |
| 4.2 | Fuzzy model-based control | 73 |
| 4.3 | Controllers with adaptive fuzzy parameters | 75 |
| 4.3.1 | Air Separation Unit | 75 |
| 4.4 | Conclusion of the chapter | 78 |
| 5 | Fuzzy MPC for an industrial autoclave | 80 |
| 5.1 | Development and implementation of a FMPC for an industrial autoclave | 81 |
| 5.1.1 | Neurofuzzy model of the temperature inside the autoclave | 82 |
| 5.1.2 | Fuzzy generalised Predictive Control (FGPC) of the temperature | 84 |
| 5.1.3 | Experimental results | 87 |
| 5.1.4 | FLC code accomplishing IEC 61131-7 | 90 |
| 5.2 | FMPC with constraints. Implementation issues | 91 |
| 5.3 | Conclusion of the chapter | 93 |
| 6 | Complexity reduction in fuzzy systems using Functional Principal Component Analysis | 94 |
| 6.1 | Complexity reduction in fuzzy systems | 94 |
| 6.2 | Principal Component Analysis | 95 |
| 6.3 | Functional Principal Component Analysis | 97 |
| 6.4 | FPCA for fuzzy systems | 99 |
| 6.5 | Illustrative examples | 101 |
| 6.5.1 | Pilot plant | 101 |
| 6.5.2 | Mechanical system | 105 |
| 6.6 | Conclusion of the chapter | 108 |
| 7 | FPCA to simplify MPC implementation | 109 |
| 7.1 | Dimensionality reduction of input variables space | 109 |
| 7.2 | Application of FPCA to FMPC without constraints | 113 |
| 7.3 | FPCA applied to PWA systems | 114 |

| | |
|--|------------|
| 7.3.1 Example: distillation column | 118 |
| 7.4 Simplified FMPC with constraints | 121 |
| 7.5 Conclusion to the chapter | 124 |
| Conclusion and future works | 125 |
| | |
| Bibliography | 127 |

Acronyms

| | |
|----------|---|
| ANFIS | Adaptive Neuro-Fuzzy Inference System |
| ANN | Adaptive Neural Networks |
| AWG | Additive White Gaussian |
| BP | Backpropagation |
| CARIMA | Controlled Auto-Regressive Integrated Moving Average |
| CSMA/CA | Carrier Sense Multiple Access/Collision Avoidance |
| DCS | Distributed Control System |
| dm | degree of membership |
| DMC | Dynamic Matrix Control |
| EA | Evolutionary algorithms |
| EHAC | Extended Horizon Adaptive Control |
| eMMMPC | explicit Min-Max MPC |
| eMPC | explicit nominal MPC |
| EPSAC | Extended Prediction Self-Adaptive Control |
| FBD | Functional Block Diagram |
| FC | Fuzzy Controllers |
| FCL | Fuzzy Control Language |
| FGPC | Fuzzy generalised Predictive Control |
| FIC | Fuzzy Incremental Controller |
| FIE | Fuzzy Inference Engine |
| FIM | Fuzzy Inference Machine |
| FIS | Fuzzy Inference System |
| FL | Fuzzy Logic |
| FLF | Fuzzy Lyapunov Function |
| FME | Fuzzy Matching Engine |
| FMPC | Fuzzy Model Predictive Control |
| FNN | Fuzzy Neural Network |
| FPCA | Functional Principal Component Analysis |
| FPD | Fuzzy Proportional Derivative |
| GA | Genetic Algorithms |
| GPC | Generalised Predictive Control |
| HMI | Human-Machine Interface |
| IAE | Integral of Absolute Error |
| IIM-CSIC | Instituto de Investigaciones Marinas Consejo Superior de Investigaciones Cientificas |
| IMC | Internal Model Control |
| ITAE | Integral of Time Absolute Error |

| | |
|--------|---|
| KKT | Karush-Kuhn-Tucker |
| LMI | Linear Matrix Inequalities |
| LTV | Linear Time Variant |
| MAC | Model Algorithmic Control |
| MAI | Multiple Access Interference |
| MF | Membership Function |
| MIMO | Multi-Input Multi-Output |
| MPC | Model-based Predictive Control |
| mpQP | multi-parametric Quadratic Programming |
| MPP | Mid Point Purity |
| NARMAX | Nonlinear Auto Regressive Moving Average with Exogenous Input |
| NMPC | Non Linear Model Predictive Control |
| NN | Neural Networks |
| P | Proportional |
| PCA | Principal Component Analysis |
| PD | Proportional Derivative |
| PDC | Parallel Distributed Compensation |
| PI | Proportional Integral |
| PID | Proportional Integral Derivative |
| PFC | Predictive Functional Control |
| PLC | Programable Logic Controller |
| PLF | Piece-wise Lyapunov Function |
| PWA | Piece Wise Affine |
| QoS | Quality of Service |
| QP | Quadratic Programming |
| RFNN | Recurrent Fuzzy Neural Networks |
| RMPC | Robust Model Predictive Control |
| RMSE | Root Mean Squared Error |
| RSSI | Received Signal Strength Indicator |
| SC | Subtractive Clustering |
| SISO | Single-Input Single-Output |
| SINR | Signal to Interference plus Noise Ratio |
| SVD | Singular Value Decomposition |
| SVM | Support Vector Machines |
| TS | Takagi-Sugeno |
| WSN | Wireless Sensor Network |

To my parents

Introduction

There are many applications of advanced control in industry. However, most of them are based on PIDs. One of the reasons for this lack of technology transfer is the difficulty of obtaining sufficiently accurate process models [1], since most of the modern techniques are model-based. Also, the level of mathematics that is required to understand the advanced control techniques, represents a drawback for the use of them by control engineers in practice [2, 3]. In addition, the complexity associated with advanced control algorithms requires high computing power, i.e. more expensive budget in facing the design of a control technique. This makes many complex industrial systems being controlled poorly. Experience says that when the advanced control engineer provides easy adjustment of the controllers designed, plant operators are more confident in using them.

Industrial implementation of advanced control systems

The implementation of advanced control systems depends on the application field. It can be found from redundant control systems of large computing capacity in process plants to low cost embedded systems applied anywhere. Due to the necessary robustness and reliability for the process industry, the implementation of advanced control is performed through a hardware platform known as *Distributed Control System* (DCS). A DCS is composed by different control units which are connected themselves to the same network. The system is organised in several levels (see Fig. 1). Although they are composed of different systems and networks, the DCSs are normally offered as one single package. The first DCS were introduced in 1975 by Honeywell and Yokogawa respectively. Historically DCSs were developed in the process industries and were the technological evolution result of adding capabilities and communications to single

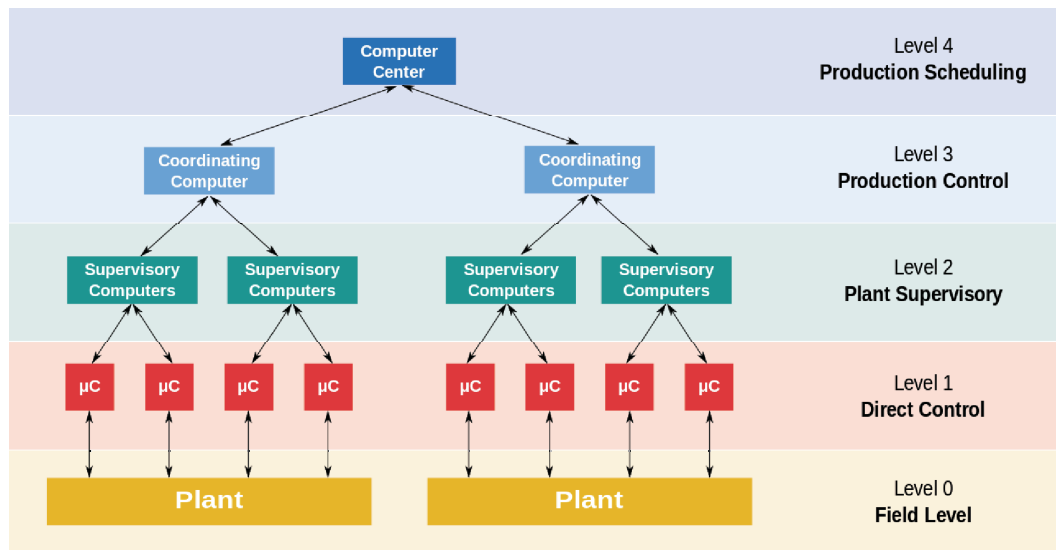


FIGURE 1: Functional levels of a typical Distributed Control System [4]

loop PID controllers and using computers for *Human-Machine Interface* (HMI) functions. Usually, many DCS systems used to use proprietary communications protocols and programming languages, but currently they are more open and flexible. On other hand, in the past it was the *Programmable Logic Controller* (PLC) the preferred system by automotive control and since then, has been robust and reliable choice for any automation system. It was introduced in 1968 by MODICON[®]. Initially, PLCs replaced relay logic systems and were programmed from proprietary panels using ladder logic. Today we find an extensive range of PLC ranging from small controllers with limited useful capacity for simple automation, to powerful computing systems suited to support many control loops. Due to developments that have suffered the hardware platforms, there is an overlap between ranges of PLCs and DCSs many doubts about choosing between them in some fields. Initially, PLC is a better choice for applications involving rapid production start using discrete I / O, while DCS has the built-in infrastructure to perform advanced regulatory control on a plant-wide scale, but the truth is that the differences today are not entirely clear. DCSs and PLCs are not mutually exclusive technologies. The classic control loops that allow stable operation of the plant are usually implemented in the DCS (or PLC). The control and instrumentation engineers are often responsible for tuning them. When an advanced control strategy is designed, the output of this type of control often takes a cascade configuration with the basic controllers who run the plant, leaving the choice of the advanced controller selection to the supervisor. In the process industry, advanced control is typically deployed on servers running as ASPEN[®] software tools, etc. or other custom tools. Now days, there is an effort to get normalised the platforms (software and hardware) used for control systems. IEC-61131

[5–13] is the international standard for PLC and IEC-61499 [14–17] for DCS.

One of the paradigms of advanced control is Model-based Predictive Control (MPC)[18]. In MPC, the controller runs a real-time optimisation to obtain the best solution for the control action. MPC uses a model of the process to predict the future evolution of the system. Setting a period of time of prediction (prediction horizon), an optimisation problem is solved to identify the best control action that minimises a cost function related to the process predictions. MPC control techniques are well known in the process industry for decades. Other fields such as aerospace, has been and is often home to develop advanced control schemes for many years. In this field, the hardware used, allows high speed computer together with a rather high strength and reliability. Today, thanks to the development of hardware platforms, advanced control is being carried out in more and more fields. The development of techniques such as model-based distributed control or obtaining explicit optimisation solutions are enabling predictive control lead to platforms low computational cost. Two important facts make it increasingly attractive application of advanced control techniques based on models. On the one hand, control of the processes associated with other fields such as building automation, smart phones, wireless sensor networks, etc., where the hardware platforms have never been known to have a high computing power, and on the other hand, increased control applications running on embedded systems, thanks to the continuous advance in hardware integration.

In this scenario, there has been and there is a constant research effort to implement the MPC hardware low capacity. The main problems affecting this purpose are on the one hand, that the MPC strategies which must be taken into account constraints, an optimisation must be solved in real time. This optimisation may require high computational cost when the prediction horizon of the model is long. On the other hand, nonlinear systems are difficult to model and the optimisation can be non-convex problem, making the real time solution difficult.

Objectives of the thesis

The main purpose of this thesis is to establish a methodology to apply nonlinear model based predictive control systems and/or subject to constraints, using as a platform, hardware systems with low computational power, allowing a realistic implementation based

on industry standards. The methodology is based on reducing the complexity of rule-based systems using the *functional principal component analysis*, providing a mathematically elegant approach to reduce the complexity of fuzzy systems and piece wise affine systems, allowing to reduce the computational load on model-based predictive control systems.

The proposal of using fuzzy inference systems, in addition to allowing nonlinear or complex systems modeling, endows of a formal structure which enables implementation of the aforesaid complexity reduction technique. Although there are many contributions made about mathematical methodology, yet heuristics and practical part of the training in fuzzy control systems and modeling. This thesis, in addition to scientific contributions discussed in the previous paragraph, describes the work done with real plants on which have been carried out tasks of modeling and fuzzy control. One of the objectives to be covered for the period of research and development of the thesis has been training with fuzzy systems and its application to industrial systems. The thesis provides a practical knowledge, based on experience. Throughout the document we can see real examples performed.

Organization of the thesis

This thesis is organised as follows: Chapter 1 starts with an overview of Model-based Predictive Control (MPC) and its application to low capability hardware systems. In Chapter 2 an introduction to fuzzy inference systems will be described. Chapter 3 is devoted to fuzzy modelling and there will be several real examples described. Here a brief description of the training techniques of fuzzy structures, choosing appropriate system inputs and various training methods was made. It will introduce different techniques of fuzzy modeling based on the experience of several years of work. Chapter 4 presents the typical control strategies based on Fuzzy systems, showing real examples carried out during the research process of the thesis. Chapter 5 presents the implementation of the fuzzy model predictive control technique, based on an real application. Chapter 6 describes the complexity reduction technique developed for Fuzzy Inference Systems, applying to real examples. This can be considered, in terms of scientific contributions, as the central chapter of the thesis. Chapter 7 describes how to simplify the complexity of predictive control systems using the techniques used in previous chapters. It shows

how you can apply the same analysis of the previous chapter on piecewise linear systems. Practical applications of complexity reduction will be seen, laying the foundations of a methodology applicable to the industry. Last chapter will conclude the thesis and will propose further developments and research.

Main contributions

This thesis focuses on the following contributions:

- A novel complexity reduction technique for fuzzy systems based on functional analysis has been study and proven experimentally.
- Application of simplification technique to allow model predictive control in low capability hardware.
- The same technique has been applied successfully for piece wise affine systems.
- There has been an application of simplification technique to allow model predictive control subject to constraints in low capability hardware, permitting an adjusting parameter at runtime.
- Transformation of inputs space for complex fuzzy models has been done in order to reduce dimensionality and therefore, complexity.
- Experimental implementation of robust model predictive control (RMPC) subject to constraints in low capability hardware. A wireless sensor network has been used to implement a energy consumption control strategy in ambulatory environment.
- Inputs selection for fuzzy modelling. Methods based on the application of component analysis and evolutionary algorithms have been carried out experimentally.
- Real applications of fuzzy control systems have been developed using standard industrial languages.
- Industrial application of fuzzy model predictive control (FMPC) has been developed and trialling.

Chapter 1

Model predictive control and its implementation

In this chapter, an overview of MPC is presented. It will not only review the formulation of various control designs that are most commonly used in industries, but practical examples will also be conducted during the investigation. The focus of this chapter is thus mainly on the practical implementation of predictive control.

1.1 Model-based predictive control

MPC consists of a set of control strategies that began to be used in the industry since the 80s [19]. The idea behind such control strategies is the optimisation of an objective function, calculating the appropriate sequence of inputs over a prediction horizon based on a model of the plant. This calculation is repeated for each sampling instant to obtain the updated information of the plant. Depending on the structure of the objective function and the type of model being used, there are several well known strategies [18]. From 1973 [20],[18], it has been shown that one of the most popular algorithms is Dynamic Matrix Control (DMC), developed by Cutler and Ramaker [21]. The process model used in DMC is the step response of the control variable:

$$y(k) = \sum_{i=1}^{\infty} g_i \Delta u(k-i) \quad (1.1)$$

The one-step ahead prediction calculated at instant k is:

$$\hat{y}(k+1|k) = g_1 \Delta u(k) + f(k+1), \quad (1.2)$$

where $f(k+1)$ is the response of $y(k+1)$ if $\Delta u(k) = 0$, i.e. the free response. The prediction over a horizon N_p is:

$$\begin{aligned} \hat{y}(k+1|k) &= g_1 \Delta u(k) + f(k+1) \\ \hat{y}(k+2|k) &= g_2 \Delta u(k) + f(k+1) \\ &\vdots \\ \hat{y}(k+N_p|k) &= \sum_{i=N_p-N_c+1}^{N_p} g_i \Delta u(k+N_p-i) + f(k+N_p). \end{aligned} \quad (1.3)$$

It can be written as a compact form,

$$\hat{\mathbf{y}} = \mathbf{G}\mathbf{u} + \mathbf{f}, \quad (1.4)$$

where

$$\mathbf{G} = \begin{pmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_c} & g_{N_c-1} & \cdots & g_1 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_p} & g_{N_p-1} & \cdots & g_{N_p-N_c+1} \end{pmatrix} \quad (1.5)$$

is the dynamic matrix.

The objective of DMC is to obtain the minimum error between the reference $w(k)$ and the control variable $y(k)$. Therefore, the manipulated variables are calculated in order to minimize a functional such that:

$$J(N_p, N_c, \lambda) = \sum_{j=1}^{N_p} \delta(j) [\hat{y}(k+j|k) - w(k+j)]^2 + \sum_{j=1}^{N_c} \lambda [\Delta u(k+j-1)]^2, \quad (1.6)$$

where the control action $u(k)$ is penalized by λ (known in the industry as *move suppression*). If there is no constraints, the sequence $\mathbf{u} = \{u(k)\}$ can be calculated by computing $\frac{\Delta J}{\Delta \mathbf{u}} = 0$, giving:

$$\mathbf{u} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{w} - \mathbf{f}). \quad (1.7)$$

Only the first control move $u(1)$ is sent to the plant and the whole process is repeated for the next sampling instant. Thus, a continuous feedback of the control variable is obtained. The prediction horizon is always the same in each sample (common thing in predictive control strategies). The difference is, with respect to any other optimal control strategy, the MPC is a *receding horizon* optimal control.

Another popular algorithm is the *generalised Predictive Control* (GPC) [22]. The output prediction in GPC is given by a CARIMA (Controlled Auto-Regressive Integrated Moving Average) model of the plant:

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t-1) + C(z^{-1})\frac{\varepsilon(t)}{\Delta}, \quad (1.8)$$

where $u(t)$ and $y(t)$ are the control and output sequence of the system, d the dead time of the plant, $\varepsilon(t)$ is a zero mean white noise,

$$\begin{aligned} A(z^{-1}) &= 1 + a_1z^{-1} + a_2z^{-2} + \dots + a_naz^{-na} \\ B(z^{-1}) &= b_0 + b_1z^{-1} + b_2z^{-2} + \dots + b_nbz^{-nb} \\ C(z^{-1}) &= 1 + c_1z^{-1} + c_2z^{-2} + \dots + c_ncz^{-nc} \end{aligned}$$

with $\Delta = 1 - z^{-1}$.

For simplicity, in the following, $C(z^{-1})$ is chosen to be 1. The sequence of future control signals is calculated such that it minimizes a multistage cost function defined by:

$$J(N_1, N_2, N_u) = \sum_{j=N_1}^{N_2} \delta(j) [\hat{y}(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(t+j-1)]^2, \quad (1.9)$$

where $\hat{y}(t+j|t)$ is a j -step ahead prediction of the system output on data up to time t , N_1 and N_2 are the minimum and maximum prediction horizon, $\delta(j)$ and $\lambda(j)$ are weighted sequences, and $w(t+j)$ is the future reference trajectory.

To solve the problem, we consider the following Diophantine equation:

$$1 = \Delta e_j(z^{-1})A(z^{-1}) + z^{-j}f_j(z^{-1}), \quad (1.10)$$

where $e_j(z^{-1})$, $f_j(z^{-1})$ are polynomials uniquely defined. They can be obtained recursively in an easy way and they can be expressed as: $e_j(z^{-1}) = e_{j,0} + e_{j,1}z^{-1} + e_{j,2}z^{-2} + \dots + e_{j,j-1}z^{-(j-1)}$,
 $f_j(z^{-1}) = f_{j,0} + f_{j,1}z^{-1} + f_{j,2}z^{-2} + \dots + f_{j,n_y}z^{-n_y}$,

In order to obtain the predictive output, let us first define the following expression:

$$g_j(z^{-1}) = \Delta e_j(z^{-1})B(z^{-1}), \quad (1.11)$$

where $g_j(z^{-1}) = g_{j,0} + g_{j,1}z^{-1} + g_{j,2}z^{-2} + \dots + g_{j,j+N_u-1}z^{-(j+N_u-1)}$,

and let us multiply equation (1.8) by $\Delta z^j e_j(z^{-1})$ to get

$$\Delta z^j e_j(z^{-1})A(z^{-1})y(k) = \Delta z^j e_j(z^{-1})B(z^{-1})u(k-d) + \bar{\xi}(k) \quad (1.12)$$

where $\bar{\xi}(k) = \Delta z^j e_j(z^{-1}) \frac{\varepsilon(k)}{\Delta}$. Therefore, by using (1.10) and (1.11), the following predictive output is obtained:

$$\hat{y}(k+j|k) = f_j(z^{-1})y(k) + g_j(z^{-1})\Delta u(k+j-d). \quad (1.13)$$

The horizon can be defined by $N_1 = d + 1$, $N_2 = d + N$ and $N_u = N$. To solve the GPC problem, the set of control signals $\mathbf{u} = [u(t), u(t+1), \dots, u(t+N)]^T$ has to be obtained in order to optimize (1.9). As the cost function is quadratic, optimum can be easily obtained, assuming there are no constraints on the control signals, making the gradient of J equal to zero. Considering $\delta(j)$ and $\lambda(j)$ constants and grouping the terms of equation (1.13) which depend on the past input and output, into \mathbf{f} , this leads to

$$\mathbf{u} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{w} - \mathbf{f}), \quad (1.14)$$

which is the same expression as (1.7). Here, $\mathbf{G} = [g_{d,0} \ g_{d+1,0} \ \dots \ g_{N_p,0}]^T$ and $\mathbf{w} = [w(t+d+1) \ w(t+d+2) \ \dots \ w(t+d+N)]^T$. The control signal that is sent to the process is the first element of \mathbf{u} , given by:

$$\Delta u(t) = \mathbf{K}(\mathbf{w} - \mathbf{f}) \quad (1.15)$$

The previous control laws (1.15) and (1.7) are calculated when there is no any constraint is taken into account. The consideration of constraints allow the process to operate closer to constraints and optimal operating conditions and may reduce the number of constraint violations, hence reducing the number of costly emergency shutdowns.

However, an online optimisation is needed. The formulation of such constrained optimal control problem can be expressed as, [18]:

$$\begin{aligned} \min_{\mathbf{u}} \left\{ J(\mathbf{u}) = \frac{1}{2} \mathbf{u}^T \mathbf{H} \mathbf{u} + \mathbf{b} \mathbf{u} + f_0 \right\} \\ \text{s. t.} \\ \mathbf{R} \mathbf{u} \leq \mathbf{r} + \mathbf{V} \mathbf{z}, \end{aligned} \quad (1.16)$$

where

$$\begin{aligned} \mathbf{H} &= 2(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I}) \\ \mathbf{b}^T &= 2(\mathbf{f} - \mathbf{w})^T \mathbf{G} \\ f_0 &= (\mathbf{f} - \mathbf{w})^T (\mathbf{f} - \mathbf{w}) \end{aligned}$$

with \mathbf{R} , \mathbf{r} , \mathbf{V} are parameters and signal bounds, and \mathbf{z} is a vector composed of present and past signals (in case of state space representation, \mathbf{z} is $x(t)$). To solve this problem, there are many available and reliable Quadratic Programming (QP) algorithms, e.g. Active Set, Feasible Direction, Pivoting methods, etc. They all use an iterative algorithm, which means that due to the computational burden they are not suitable for every hardware platform. In [23], the implementation aspects are presented and the restriction of the horizon on limited-resource hardware such as fixed-point arithmetics is addressed. Faster on line optimisation technique is described in [24, 25].

1.2 Industrial implementation of MPC

MPC implementation has always been associated with commercial products in industry [20, 26, 27]. Companies like Adersa, Aspen Tech., Honeywell, Shell Global, Invensys, Continental Controls, DOT Products, Pavilion Technologies, etc., offer their platform externally. Most of them use an objective similar to (1.16). However, for fast process or large number of variables, QP may not be sufficient. Some of the commercial platform offer suboptimal approximated optimisation algorithms for achieving that purpose. In addition to the DMC and GPC (seen above), other algorithms are well known and widely used in industry: Model Algorithmic Control (MAC)[28], Predictive Functional Control (PFC)[29], Extended Prediction Self-Adaptive Control (EPSAC) [30], Extended Horizon Adaptive Control (EHAC) [31].

Although DCS are adequate for the implementation of MPC controllers, nowadays,

there are PLCs ready to run MPC algorithms, like Siemens (unconstraint DMC)[32] and Schneider Electric (PFC)[33].

On the other hand, it is very important to offer the plant supervisors or instrumentation engineers the adjustment parameters of the controllers. As we have seen in the previous sections, these parameters are the prediction horizon and control and weights in the objective function. Several authors have proposed practical tuning methods based on real experience [34–36], [37, 38] including some tuning techniques based on trial and error. A method based on principal component selection is used in [39, 40]. [41] shows a simplification of a DMC tuning. Tuning method focusing on stability [42–45] and robustness [46, 47] are also proposed. Techniques to evaluate the performance of the controllers are presented in [48–50]. A good review of the techniques is presented in [51]. From the point of view of the implementation, prediction and control horizons are not suitable for being used as setting parameters. They should be chosen, like the sample time, depending on the process dynamic. The weights used in the formulation of DMC and GPC modulate the error terms (δ) and control action (λ), respectively. The parameter that mostly impacts the robustness is the move suppression (λ)[34]. In practice, it is very important to give the plant supervisor and process instrumentation and control engineers the chance to adjust this parameter.

1.3 MPC for low-cost hardware: a practical implementation perspective

For linearly constrained MPC problems of a low dimensional system, one can partially avoid the computational burden by precomputing the solution of the optimisation using multi-parametric quadratic programming (mpQP) [52–55]. This leads to an offline MPC. Examples can be found in [56, 57], where a constrained MPC problem is developed using a low range PLC. Also in [58], an explicit MPC is applied in building controllers. Following this line, an application will be presented in this section regarding to MPC that is implementable on a hardware with low capacity such as Wireless Sensor nodes (*moten*). A constrained explicit generalised predictive control for ambulatory wireless sensor network power management will be described.

1.3.1 Ambulatory sensor network description

This application is framed within a context that control engineering can aid in the energy sensitive provision of vital biometric data, [59]. An ambulatory wireless sensor network (WSN) scenario is considered within a typical ambient healthcare setting illustrated by Figure 1.1[60–62]. This network consists of a set of low cost (in this particular case, Mote type), communicating sensor nodes featuring low-power radios. In each health-care unit, there is a remote base station that is used to aggregate biometric data and to process the flow of information locally through wireless links. The base station is expected to relay the data wirelessly to another base station, e.g., from base station 2 (BS_2) to base station 1 (BS_1), when a patient moves to an adjoining area in an ambulatory fashion. It is anticipated that the sensor data will be routed from there via an Internet connection to some higher supervisory level within the network, [63, 64]. The nodes

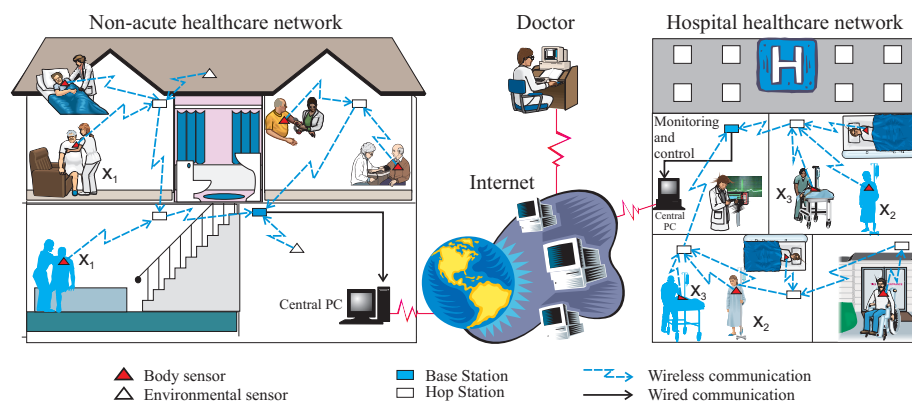


FIGURE 1.1: An ambulatory WSN in a general ambient healthcare environment.

are deployed in either a *static or mobile* manner. The variability in wireless link quality for a static network deployment, e.g., in environmental, agricultural, or structural monitoring applications, has been shown (empirically) to place significant constraints on information throughput, [65, 66]. The introduction of wearable devices, organized within an ambulatory setting is well known to exacerbate this problem, [67–69]. The communication link between any transceiver pair within such a network deployment is known to be time-varying, non-linear; and can exhibit large, rapid deviations that are dependent on placement and body movement, [70]. The resulting network must therefore be able to withstand complex radio dynamics such as fast and shadow fading, as well as relatively variable antenna orientation between transceiver pairs [71, 72], that can attenuate the received signal power at the base station, frequently leading to packet

errors (vital health status information to be lost) or even total breakdown in communication. The loss of data during wireless transfer becomes particularly more significant in this class of application than most other types of monitoring data. The primary performance requirement in the healthcare application space is to reliably achieve a target level, i.e., quality reference, on the received signal strength indicator (RSSI) metric, [73], so that a health care provider can be assured that the link quality and continuous link connectivity are sufficient to guarantee satisfactory levels of subject observation.

Commercial sensor node platforms are now generally available that support wearable devices within a body sensor network setting. Examples include the MicaZ and Telos motes used in the CodeBlue project [74] based at Harvard University, that operate with low-power radios and are based on an adoption of the IEEE 802.15.4 standard, employing a carrier sense multiple access/collision avoidance (CSMA/CA) technique for data transmission. However, the CSMA/CA mechanism does not work perfectly due to the so called “hidden terminal” or multiple access interference (MAI) problem, [75]. This interference due to the existence of multiple simultaneously transmitting nodes can unnecessarily increase transmit power level and significantly degrade network capacity. In addition, the phenomena of uncertain fading channel and interference means that the question of dynamic energy management is a challenging component of any WSN that is constrained by finite battery resources. Certainly, transmission at lower power levels will compromise the quality of communication, and the desired quality of service (QoS) might not be met. An outage-based QoS constraint is considered wherein the received signal strength must be kept above a given threshold level so that no instances of potentially catastrophic outage or disconnection due to deep fading occur [76]. It is a particular objective of this work to highlight how dynamic control can aid the development of practicable radio power control strategies that provide an intelligent way of determining the optimal transmit power levels to be used by the sensor nodes on a network. The result will be an improvement in overall network lifetime and the maintenance of an acceptable received signal strength for all sensor nodes, while also preserving a satisfactory QoS for as many nodes as possible on the network. Furthermore, the use of commercial sensor node platforms that characterises this work means that only limited computational capacity and memory are available in the implementation stage of the radio power control law.

The sensor nodes are programmed to send sensor data framed in an 802.15.4 format, [77–79]. Figure 1.2 illustrates the typical problem that has been considered. In this testbed, a star topology is adopted: a Tmote Sky sensor node is connected to a personal computer acting as the fixed base station or coordinator via the USB port, and there

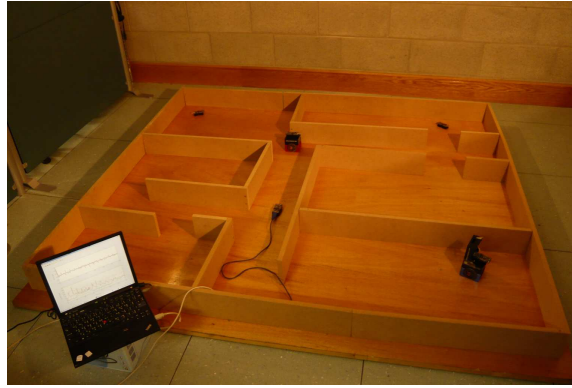


FIGURE 1.2: An experimental ambulatory test scenario containing two static and two mobile nodes.

are four other Tmote Sky sensor nodes that are wirelessly connected to, (and randomly located around), the base station within 200 cm distance. Reconfigurable obstacles that highlight the practical phenomena influencing the loss of line-of-sight between the base station and the sensor nodes are integrated to form part of the testbed as illustrated in Figure 1.2. In addition, a selection of fully autonomous MIABOT Pro miniature mobile robots are used to provide a controlled, ambulatory dimension to the experiment. Each of the robots can be mounted with the sensor node in order to imitate various activities performed by a patient. The data flow within the WSN testbed is depicted in Figure 1.3. In this work, power control algorithms are directly implemented (via nesC) on the Tmote Sky sensor node platform running the TinyOS operating system, [80], that is identified *a priori* as the coordinator. We chose this environment, since the TinyOS and the Tmote Sky architectures are *de facto* benchmark standards for commercial and academic WSNs. An interface between Matlab and TinyOS has been established using stable bridging tools written in Java for data management purposes. Upon receipt of data packets from each sensor node at time sample t , the coordinator takes an RSSI measurement and then performs the power control strategy, resulting in an optimal power increment desired for updating the actual transmit power for the next sampling instant. After the hardware constraints (i.e., quantisation and power limitation) of the radio power amplifier are taken into account, the quantised power level is then transmitted over a fading channel to the corresponding connected sensor, where the RF output power level is adjusted accordingly.

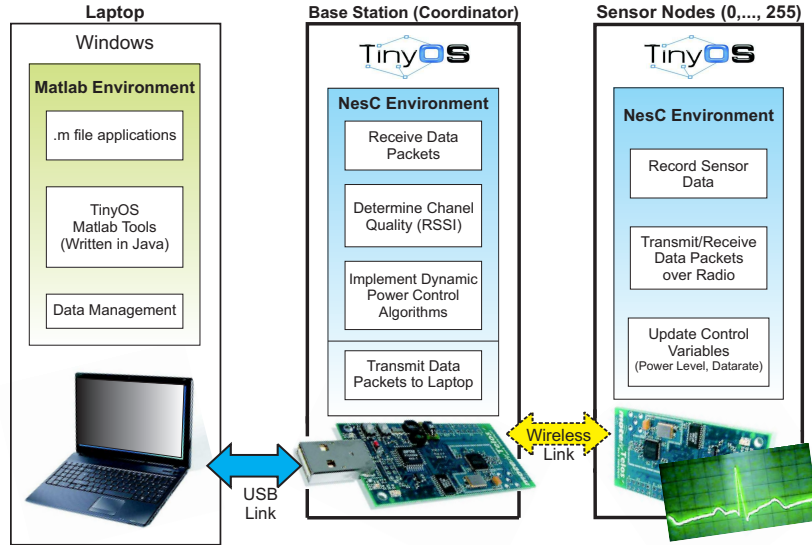


FIGURE 1.3: Data flow within the IEEE 802.15.4 WSN testbed.

1.3.2 Problem formulation

We consider an ambulatory sensor network composed of one base station and n sensor nodes, labeled $i = 1, \dots, n$, connected in a star topology, in which each sensor node i transmits using a power level $p_i > 0$, $i = 1, \dots, n$. At the base station, the signal is despread, demodulated, and decoded in order to get the source data. The achieved *signal-to-interference-plus-noise-ratio* (SINR) at time k , for the generic transceiver pair i , is given by

$$\gamma_i(k) = \frac{g_i(k)p_i(k)}{I_i(k)} = \frac{g_i(k)p_i(k)}{\sum_{j \neq i} g_j(k)p_j(k) + \eta_0}, \quad (1.17)$$

where channel gain g_i represents the attenuation in the radio link due to path loss, log-normal shadowing, and Rayleigh fading between the i -th sensor node and the base station, I_i denotes the MAI caused by other transmitting nodes plus the noise power at the base station $\eta_0 > 0$, which is assumed to be additive white Gaussian (AWG). Transforming (1.17) into the logarithmic domain results in¹

$$\bar{\gamma}_i(k) = \bar{p}_i(k) + \bar{g}_i(k) - \bar{I}_i(k). \quad (1.18)$$

In general, wireless sensors, and in particular the Tmote Sky sensor nodes, do not provide an explicit SINR as a link performance metric, but instead an RSSI measurement

¹Throughout the paper, the logarithmic (e.g. dB or dBm) value of a variable x in linear scale is denoted by \bar{x} , namely, $\bar{x} = 10 \log_{10} x$.

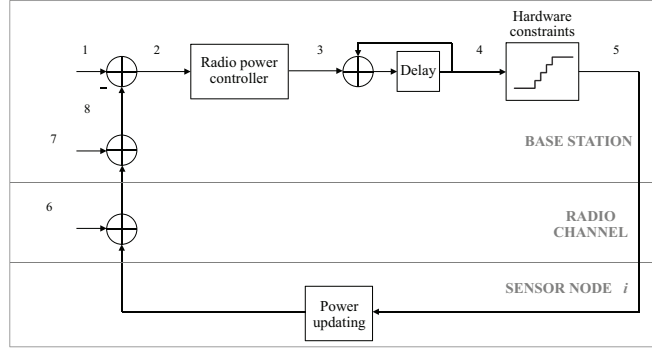


FIGURE 1.4: A closed-loop decentralised power control system when applying the general power control algorithm.

is given. The RSSI has been shown to have a strong relationship with the SINR in this problem setting, see [81], and the use of the RSSI as a performance measure for communication for power control purposes is now a feature of the literature, see [68, 81–83].

An estimate of the SINR in dB from the RSSI in dBm established in [81] is given by

$$\bar{\gamma}_i(k) \approx \bar{r}_i(k) - C - 30, \quad (1.19)$$

where \bar{r}_i denotes the RSSI of sensor node i , C represents measurement offset, and the term 30 accounts for the conversion from dBm to dB.

Substituting the SINR $\bar{\gamma}_i$ (1.18) in (1.19) yields the following relation,

$$\bar{r}_i(k) = \bar{p}_i(k) + \bar{g}_i(k) - \bar{I}_i(k) + C + 30. \quad (1.20)$$

At time k , the base station measures the RSSI upon receipt of data packets transmitted by the i -th sensor node, \bar{r}_i . Once the RSSI is measured, it is used to compare with an RSSI target \bar{r}^t for each node at the base station to obtain the RSSI tracking error \bar{e}_i ,

$$\bar{e}_i(k) = \bar{r}^t - \bar{r}_i(k). \quad (1.21)$$

The control error is fed into the power controller, which executes the power control algorithm. The control law determines the power control update command \bar{u}_i to force \bar{r}_i to follow the target. Then, upon reception of \bar{u}_i at the power multiplier, the power \bar{p}_i to transmit the next data packets is set by

$$\bar{p}_i(k+1) = \bar{p}_i(k) + \bar{u}_i(k). \quad (1.22)$$

Before the updated transmit power is sent via the channel to the sensor node, it is quantized to a finite discrete power level \bar{p}'_i , which is then used by the sensor node's radio power amplifier. In the transformation process, a quantisation error \bar{q}_i is introduced, i.e., $\bar{p}'_i = \bar{p}_i + \bar{q}_i$.² In addition, the sensor node power level assigned by the base station that is subject to the limitation of power amplifier output can be saturated³. The transmitted signal is additionally corrupted by highly time-varying uncertain interference, noise, and channel gain. Then, we have that the RSSI (1.20) can be rewritten as

$$\bar{r}_i(k) = \bar{p}_i(k) + \bar{q}_i(k) + \bar{s}_i(k) + \bar{g}_i(k) - \bar{I}_i(k) + C + 30, \quad (1.23)$$

where \bar{s}_i is the nonlinear effect of power amplifier saturation. A simplified closed-loop interpretation of the proposed decentralised RSSI-based power control system is shown in Fig. 1.4.

Combining (1.21)-(1.23), the uncertain linear system representation for the RSSI tracking error of sensor i is obtained as

$$\bar{e}_i(k+1) = \bar{e}_i(k) - \bar{u}_i(k) + \bar{w}_i(k), \quad (1.24)$$

The disturbance \bar{w}_i accounts for the complex time-varying network dynamics of the power control process. For now, consider perturbations null, taking as a nominal model:

$$\bar{x}_i(k+1) = \bar{x}_i(k) - \bar{u}_i(k+1) \quad (1.25a)$$

$$\bar{e}_i(k) = \bar{x}_i(k), \quad (1.25b)$$

where \bar{x}_i represents the state of the system. In order to introduce the effect of feedback in the predictions, see e.g., [85, 86], we assume that the control input is defined as follows,

$$\bar{u}(k) = -K\bar{x}(k) + \bar{v}(k), \quad (1.26)$$

where K is a linear gain. This implies that the control moves \bar{u} for updating the transmit power are corrected by \bar{v} which are computed by the MPC controller.

²Empirical evidence suggests that the practical effect of such a quantization error in power control of WSNs is in fact negligible with respect to the other sources of uncertainty, i.e., channel fading and interference effects, [83].

³This hardware limitation is a fact of life for any commercial radio chipset such as CC2420, [84].

1.3.3 MPC

The MPC will solve the following constrained optimisation problem:

$$\begin{aligned}
 J^*(\bar{x}) &= \sum_{j=0}^{N_p-1} [\bar{x}(k+j|k)^T Q \bar{x}(k+j|k) + \bar{u}(k+j|k)^T R \bar{u}(k+j|k)] + \bar{x}(k+N_p|k)^T P \bar{x}(k+N_p|k) \\
 \text{s.t. } & \bar{x}(k+j|k) \in \bar{X}, j=0, \dots, N_p; \quad \bar{u}(k+j|k) \in \bar{U}, j=0, \dots, N_p-1
 \end{aligned} \tag{1.27}$$

where N_p is the prediction horizon, $\bar{x}(k|k) = \bar{x}$ is the initial state, $\bar{x}(k+j|k)$ and $\bar{u}(k+j|k)$ are the predicted state and control input, respectively, $\bar{\mathbf{v}} = [\bar{v}(k|k)^T, \dots, \bar{v}(k+N_p-1|k)^T]^T$ is the sequence of correction control inputs, \bar{X} and \bar{U} are polyhedra defined by the state and input constraints respectively. $Q \geq 0$, $P \geq 0$, and $R > 0$ are weight to permit adjustment of the controller. the explicit solution $\bar{\mathbf{v}}^*(\bar{x})$ of the min-max mpQP problem is defined as a continuous piecewise-affine (PWA) function characterised over a polyhedral partition of the feasible set of state $S_F = \{(\bar{x}, \bar{\mathbf{v}}) | F\bar{x} + G\bar{\mathbf{v}} \leq d\}$,

$$\bar{\mathbf{v}}^*(\bar{x}(k)) = \begin{cases} K_1 \bar{x}(k) + q_1, & \text{if } \bar{x}(k) \in \bar{\mathcal{X}}_1 \\ K_2 \bar{x}(k) + q_2, & \text{if } \bar{x}(k) \in \bar{\mathcal{X}}_2 \\ \vdots \\ K_{N_{rej}} \bar{x}(k) + q_{N_{rej}}, & \text{if } \bar{x}(k) \in \bar{\mathcal{X}}_{N_{rej}}, \end{cases} \tag{1.28}$$

with a polyhedral partition $\mathcal{P} = \{\bar{\mathcal{X}}_1, \dots, \bar{\mathcal{X}}_{N_{rej}}\}$, where

K_i and q_i are respectively the control gain and offset for each region, and N_{rej} is the number of regions. If the state lies outside the feasibility region, $\bar{\mathbf{v}}$ is set to zero, which usually saturates the control input.

Therefore, the application of an explicit MPC for optimal power assignment is limited to evaluating the piecewise-affine function and can be written as the following algorithm:

Algorithm:

1. At sample k , get the current state $\bar{x}(k)$.
2. Perform a sequential search: if $\bar{x}(k) \in \bar{\mathcal{X}}_i$, then $\bar{v}(k) = K_i \bar{x}(k) + q_i$ is given for the specific region i , $i \in \{1, \dots, N_{rej}\}$.
3. Then, apply $\bar{u}(k) = -K \bar{x}(k) + \bar{v}(k)$ for the power control update command.
4. Get new state measurement, and repeat the search at sample $k+1$.

1.3.4 Robust MPC

Taking now into account uncertainties, the model (1.25) turns into:

$$\bar{x}_i(k+1) = \bar{x}_i(k) - \bar{u}_i(k+1) + \bar{w}_i(k) \quad (1.29a)$$

$$\bar{e}_i(k) = \bar{x}_i(k), \quad (1.29b)$$

The disturbance \bar{w}_i that enters the power control loop cannot be measured and is also difficult to estimate. A feature of the MPC design paradigm is that such disturbance effects can be expressed using a so-called min-max formulation, [87], which can be expressed as

$$\begin{aligned} J^*(\bar{x}) &= \min_{\bar{\mathbf{v}}} \max_{\bar{\mathbf{w}} \in W_{N_p}} V(\bar{x}, \bar{\mathbf{v}}, \bar{\mathbf{w}}) \\ \text{s. t. } \quad &\bar{x}(k+j|k) \in \bar{X}, \forall \bar{\mathbf{w}} \in W_{N_p}, j = 0, \dots, N_p \\ &\bar{u}(k+j|k) \in \bar{U}, \forall \bar{\mathbf{w}} \in W_{N_p}, j = 0, \dots, N_p - 1, \end{aligned} \quad (1.30)$$

where $\bar{\mathbf{w}} = [\bar{w}(k)^T, \dots, \bar{w}(k+N_p-1)^T]^T$ represents a possible sequence of input disturbances to the system, $W_{N_p} \subseteq \mathcal{R}^{N_p \cdot n_{\bar{w}}}$ denotes the set of possible disturbance sequences of length N_p , $W_{N_p} = W \times W \times \dots \times W$, where \times denotes the cartesian product and $V(\bar{x}, \bar{\mathbf{v}}, \bar{\mathbf{w}})$ is the objective function defined as

$$\begin{aligned} V(\bar{x}, \bar{\mathbf{v}}, \bar{\mathbf{w}}) &= \sum_{j=0}^{N_p-1} [\bar{x}(k+j|k)^T Q \bar{x}(k+j|k) \\ &\quad + \bar{u}(k+j|k)^T R \bar{u}(k+j|k)] \\ &\quad + \bar{x}(k+N_p|k)^T P \bar{x}(k+N_p|k), \end{aligned} \quad (1.31)$$

Min-max optimisation problems in general exhibit a very high computational complexity. In [88], it is shown that the min-max MPC problem (1.30) can be reformulated into an mpQP problem, [88], Being able to solve as was seen in 7.30

The tuning parameters used for deriving the explicit min-max MPC controller are as follows: $N_p = 1$, $Q = P = R = 1$, and $K = -0.618$. The state and control input constraints are given as $-15 \leq \bar{x} \leq 5$ and $|\bar{u}| \leq 10$, respectively. Note that the state constraints are adopted from an explicit consideration of the RSSI threshold value that is required. Such tuning parameters, i.e. the cost matrices, of the MPC controller were also obtained by extensive simulations.

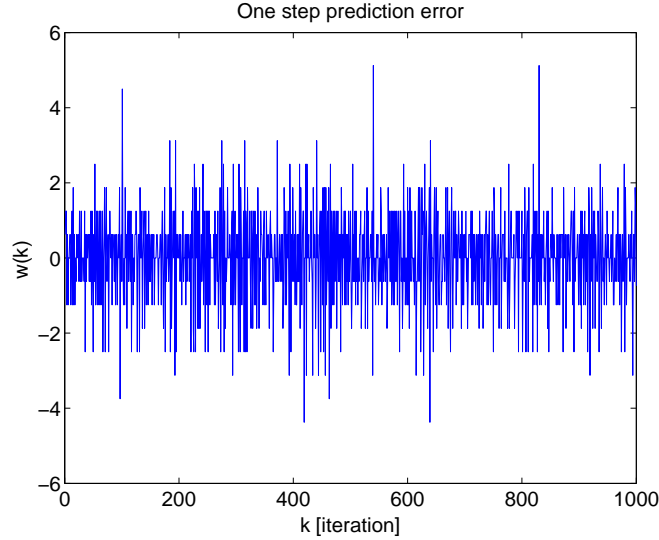


FIGURE 1.5: One-step prediction error.

The model used by the controller is given by (1.29), in which the only parameter to estimate is the uncertainty bounds. To characterise the uncertainty bounds, we have carried out an experiment, in which four sensor nodes were deployed within an ambulatory setting, i.e., two static and two mobile nodes, and performed a nominal MPC-based power control (see below). The uncertainty bounds have been set according to the error computed using the predicted state, which came from the prediction model (1.24), where $\bar{w} = 0$. The errors⁴ computed using this model are shown in Fig. 1.5. The uncertainty bounds have been set to -2.5 and 2.5, since 92.53% of the errors in the test set are within these bounds.

The solution of the min-max MPC mpQP problem, obtained from the model (1.24), the tuning parameters, the constraints, and the uncertainty bounds, provides a polyhedral partition over the \bar{x} -space, consisting of 3 regions. The algorithm that finds the optimal control correction effort $\bar{v}^*(0)$ reduces to the following lookup table that depends on the current state of the RSSI tracking error,

$$\mathbf{if} \begin{bmatrix} -0.071 \\ 1 \end{bmatrix} \bar{x}(k) \leq \begin{bmatrix} 0.5 \\ 5 \end{bmatrix} \mathbf{then}$$

$$\bar{v}^*(0) = 0.382 * \bar{x}(k)$$

$$\mathbf{else if} \begin{bmatrix} 11.025 \\ -0.559 \end{bmatrix} \bar{x}(k) \leq \begin{bmatrix} -78.125 \\ 6.875 \end{bmatrix} \mathbf{then}$$

⁴The errors plotted in Fig. 1.5 were chosen from the worst state prediction errors provided by one of the two mobile nodes.

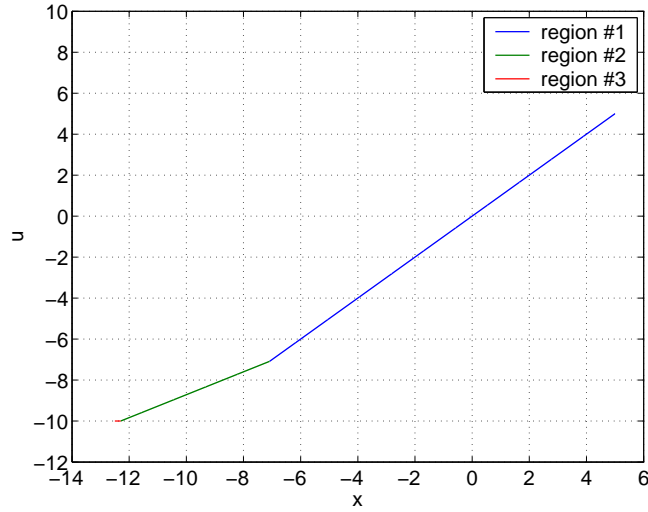


FIGURE 1.6: Min-max MPC-based radio power control law.

$$\bar{v}^*(0) = -0.059 * \bar{x}(k) - 3.125$$

$$\mathbf{else\ if} \begin{bmatrix} 2.236 \\ -1 \end{bmatrix} \bar{x}(k) \leq \begin{bmatrix} -27.5 \\ 12.5 \end{bmatrix} \mathbf{then}$$

$$\bar{v}^*(0) = -0.618 * \bar{x}(k) - 10$$

else {problem is infeasible}

$$\bar{v}^*(0) = 0$$

end if

It can be seen that when the optimal control law is saturated, the control correction input \bar{v} has the same gain as the feedback law, i.e., $K = 0.618$, but is of inverted sign, so that the applied input is therefore independent of the state of the system. As a result, the min-max MPC-based radio power control law $\bar{u}^*(0)$ generates the profile illustrated in Fig. 1.6.

1.3.5 Experimental results

The min-max MPC mpQP solution computed requires a memory usage of only 285 bytes from an available 48 Kbytes of memory provided within a typical Tmote Sky

TABLE 1.1: Summary of test scenarios.

| Test scenarios | Sensor nodes deployment | Motion of the mobile robot(s) |
|----------------|-----------------------------------|---|
| 1 | 3 static nodes, 1 mobile node | 1 robot — straight path |
| 2 | 2 static nodes, 2 mobile nodes | 1 robot — straight path 1 robot — circular path |
| 3 | 1 static node, 3 mobile nodes | 2 robots — straight path 1 robot — circular path |

sensor node, [89]. There still exists space for various *actual* user level sensor node applications. Note that the number of regions depend on the complexity of the model and the prediction horizon. In theory, the possible number of regions can grow exponentially with the horizon dimension. However, for low order systems this number is much lower because of the state constraints, [90, 91].

To benchmark the advantages of the proposed algorithm, the explicit min-max MPC-based radio power controller (denoted henceforth as eMMMPC) is compared in each scenario with the explicit nominal MPC approach seen before (denoted as eMPC), a power control that utilizes a balanced adaptive scheme (denoted as Adaptive 1) and has previously been shown to perform well in the WSN power control literature [68] as well as a power control strategy with adaptive step-size (denoted as Adaptive 2), taken from [92].

In order to benchmark the proposed control law with the other aforementioned designs, three test scenarios, in which the sensor nodes are deployed in both a static and mobile fashion, have been considered and the setup details are summarised in Table 1.1. Specifically, in scenarios 2 and 3 the motion of robot is configured to move continuously in a circular path about its initial position, along a short distance of 20 cm, so as to produce a high variation in the observed RSSI feedback signal. The wireless channel can hence be well described as exhibiting a Rayleigh fading distribution in this circumstance, [81]. For each scenario, the test has been iteratively performed for 4 runs with a duration of 140 sec and a sampling interval of 1 sec. For each run, different static nodes positions and mobile nodes trajectories (either straight or circular path) are defined. Note that, for consistency, the positions of static nodes and the trajectories of the mobile robots are not changed while an experiment is repeated for each control law. During each experiment, dynamic data, i.e., RSSI and the corresponding transmit power level of all nodes are recorded to analyze the system performance according to the following three criteria:

- Power consumption:

$$P_i(\text{mW}) = \sum_{k=1}^{N_s} p_i(k), \quad (1.32)$$

where N_s is the total number of samples.

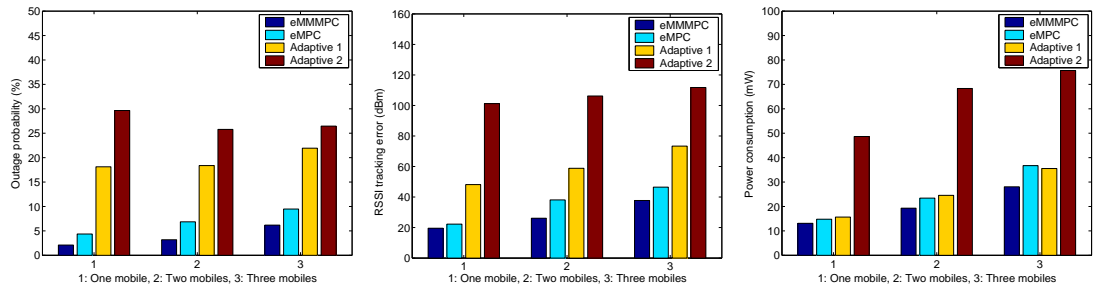
- Outage probability:

$$P_{o_i}(\%) = \text{Prob}\{\bar{r}_i < \bar{r}^{\text{th}}\}. \quad (1.33)$$

- Standard deviation of the RSSI tracking error:

$$\sigma_{e_i}(\text{dBm}) = \left(\frac{1}{N_s} \sum_{k=1}^{N_s} (\bar{r}^t - \bar{r}_i(k))^2 \right)^{\frac{1}{2}}. \quad (1.34)$$

The mean values of P_i , P_{o_i} , and σ_{e_i} of all nodes are calculated for each experiment of each controller. The averages of these results for each test scenario are the final performance metrics of each controller.



(a) Average outage probability (b) Average standard deviation of the RSSI tracking error (c) Average power consumption

FIGURE 1.7: Average values of outage probability, standard deviation of the RSSI tracking error, and power consumption for different radio power controllers of all sensor nodes.

Figure 1.7 and Table 1.2 present the summary of performance evaluation in terms of the average power consumption \bar{P} , the average outage probability \bar{P}_o , and the average standard deviation of the RSSI tracking error $\bar{\sigma}_e$ produced for different test scenarios according to the performance criteria (1.32), (1.33), and (1.34), respectively. More detail can be seen in [62].

TABLE 1.2: Performance evaluation for different radio power controllers taking into account all sensor nodes.

| | | eMMMPC | eMPC | Adaptive 1 | Adaptive 2 |
|---------------|--------------------------|--------|-------|------------|------------|
| One mobile | \tilde{P}_o (%) | 2.10 | 4.36 | 18.12 | 29.65 |
| | $\tilde{\sigma}_e$ (dBm) | 19.55 | 22.24 | 48.16 | 101.22 |
| | \tilde{P} (mW) | 13.11 | 14.77 | 15.65 | 48.66 |
| | | eMMMPC | eMPC | Adaptive 1 | Adaptive 2 |
| Two mobiles | \tilde{P}_o (%) | 3.19 | 6.86 | 18.38 | 25.78 |
| | $\tilde{\sigma}_e$ (dBm) | 26.11 | 38.12 | 58.84 | 106.20 |
| | \tilde{P} (mW) | 19.31 | 23.41 | 24.57 | 68.32 |
| | | eMMMPC | eMPC | Adaptive 1 | Adaptive 2 |
| Three mobiles | \tilde{P}_o (%) | 6.18 | 9.48 | 21.93 | 26.46 |
| | $\tilde{\sigma}_e$ (dBm) | 37.74 | 46.48 | 73.37 | 111.76 |
| | \tilde{P} (mW) | 28.05 | 36.72 | 35.53 | 75.68 |

1.4 Conclusion to the chapter

In this chapter, an introduction of MPC and its implementation have been presented. An overview of tuning techniques for MPC commonly used in the industrial process is also addressed. From a practical point of view, a RSSI-based explicit min-max MPC approach has been presented to address the radio power control problem encountered in ambulatory sensor networks. It has been shown that an explicit solution of the constrained min-max MPC problem can be computed for the WSN power control problem by solving an mpQP. The feasibility of the proposed design and its performance has been experimentally validated using a variety of test scenarios. The experimental results clearly show that the explicit min-max MPC-based power management strategy performing optimal radio power assignments exhibits good performance for this particular problem. Moreover, the algorithm has been implemented on a reduced functionality wireless sensor platform and the numerical overhead involved is relatively insignificant from an implementation perspective. However, although the strategy is suitable for industrial processes (process industry, manufacturing, etc.), lack of on-line tuning parameters to the designed controller is a weakness in terms of implementation. One possible solution is to design different PWA controllers for different settings parameters and linearly interpolating the longline action of the different controllers. The problem with this is an increase in complexity. In the following chapters, we will address this problem by using Fuzzy models and a complexity reduction technique.

Chapter 2

Implementation of fuzzy inference systems

This introductory chapter will give an overview about the foundation of *fuzzy logic* (FL) and its applications. Now days, FL is the paradigm of the Soft Computing, discipline of computer science which integrates a set of techniques that deal with approximation, imprecision, uncertainty and partial truth. (such as neural networks, evolutionary computation, support vector machines, etc.). Many engineering applications have been developed based on the use of fuzzy logic [93]. Since 1965, when L. Zadeh published his article about fuzzy sets, the count of publications containing the word “fuzzy” in title, as cited in INSPEC database is 171.420 and in MathSciNet database, 27.201 (Compiled on July 26, 2015). The total number of papers with “fuzzy” in title in Google Scholar is 2.310.000. There are 29 journals with ”fuzzy” in the title (and 21 with ”soft computing”). And there are 1255 patents issued with ”fuzzy logic” into the title and 2.314 with ”fuzzy control”, according to *The Lens* (<http://www.lens.org>).

2.1 Fuzzy logic

FL is a logical system built on the basis of *fuzzy sets*, introduced by Lotfi A. Zadeh[93]. According to him, if $X = \{x\}$ is a space of objects (points) a fuzzy sets A in X is characterised by a function $f_A(x) : x \rightarrow [0, 1]$, called *membership function*. In classic set theory the membership function can take only two possible values: $\{0, 1\}$. FL is, therefore, a multivalued logic. Fuzzy and classic inference are to obtain a conclusion statement

from another (premise) by applying inference rules, they only differ on belonging to sets of variables used in the antecedents and consequences. Fuzzy logic can handle information closer to the human way, ie, uncertain, vague or imprecise. A plant operator is able to control a system using logic, running perfectly control actions not defined by numbers, such as: close valve *slightly*, *slightly* slow down or set to a *fairly high* temperature. The operator controls with *significance* more than accurately.

Modus Ponens inference rule is used in applications of logic in engineering because it preserves the cause-effect. *Modus Ponens* or direct reasoning can be summarized as follows:

- Premise 1: x is A
- Premise 2: IF x is A , THEN y is B
- Consequent: y is B

Modus Ponens is associated with the implication $A \rightarrow B$. As for the theory of fuzzy sets, also the foundations of the theory of fuzzy logic depart and take the fundamental concepts of classical logic. In fuzzy logic, *modus ponens* extends to *generalised modus ponens*, where the antecedents (premise 1 and 2) and consequent are activated by a degree of membership $\in [0, 1]$. Generally, a fuzzy inference system (FIS) is structured as shown in fig 2.1. The fuzzification is determined by the degrees of membership of the inputs to the sets mentioned on the premises. In order to formulate mathematically the fuzzy inference system a mechanism has to be defined for the operators, implication, aggregation and defuzzification. There are two types of well known fuzzy systems:

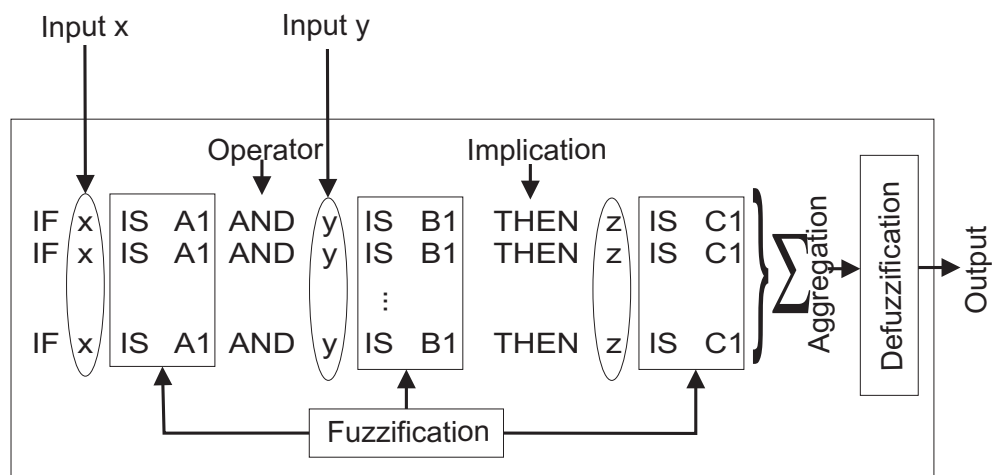


FIGURE 2.1: Fuzzy inference system structure

Mamdani-type and Takagi-Sugeno type.

2.1.1 Mamdani fuzzy systems

Mamdani System is a structure proposed by E. Mamdani [94] in 1975. A Mamdani fuzzy system consists of fuzzy antecedent and consequent, a rule set, a inference engine that processes the reasoning process using the rules set. The idea of this structure is to carry out an approximation of the human reasoning. A set of conditional rules (IF...THEN...) connects the inputs with the outputs. Those rules work with fuzzy predicates. Given the inputs (non-fuzzy), the fuzzification is carried out by the evaluation of prototypical memberships functions (fig. 2.2) The degree of membership (dm) is a value

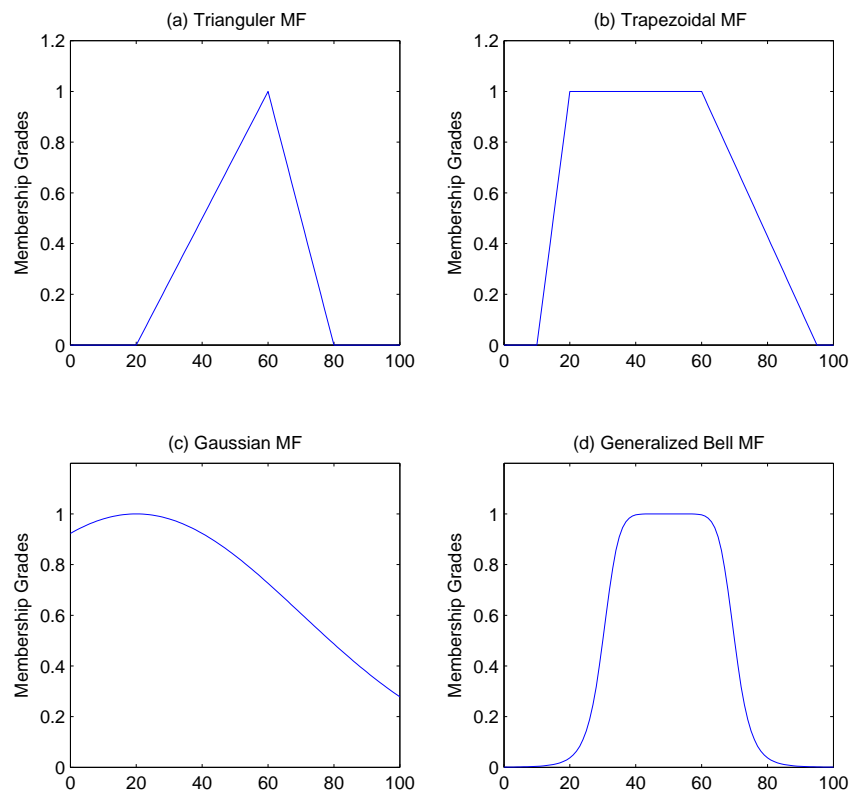


FIGURE 2.2: Common Membership functions

between 0 and 1 assigned to a non-fuzzy number and gives it the degree of belonging to a particular fuzzy set represented by a membership function. In other words,

Definition 2.1. Let X be an universal set, and a fuzzy subset A of X . The membership function of A is defined as:

$$\mu_A : X \rightarrow [0, 1]$$

the value $\mu_A(x)$ is the degree of belonging of x to the fuzzy set A . Once the fuzzification is made and the dm to every fuzzy system are calculated, they will decide the degree of activation of each rule. In case of multiple inputs, an operation between degrees of membership must be performed in order to obtain the dm of the antecedent. Multiplication or minimum of dm could be chosen if they are connected by AND operator, and summation or maximum of dm if OR operator. The result of the operation, will decide the degree of firing of the particular rule. A way to define the numerical firing of a rule is by using α -cuts.

Definition 2.2. Let X be an universal set, and a fuzzy subset A of X . Let $\alpha \in \mathbb{R}$ and $\alpha \in [0, 1]$. then, the α -cut of A is defined as:

$$\alpha(A) = \{x \in X | \mu_A(x) \geq \alpha\}$$

If the activation degree of rule i is μ_i and the fuzzy consequent is the fuzzy set $U_i(x)$, defined by the membership function $\mu_{U_i}(x)$, then, if α_i is the α -cut of $U_i(x)$, with $\alpha = \mu_i$, the firing degree of such rule is defined by the function:

$$\bar{U}_i(x) = \begin{cases} \mu_i & \text{if } x \in \alpha_i \\ \mu_{U_i}(x) & \text{if } x \notin \alpha_i \end{cases} \quad (2.1)$$

Taking the $\bar{U}_i(x)$ function for each rule, all of them are combined into a single fuzzy set. Normally, set is formed by the union of the new membership functions of the outputs $u(x) = \bigcup \bar{U}_i(x)$, so called *aggregated fuzzy set*. There are several methods of defuzzification to obtain the nonfuzzy real output. The most popular defuzzification method for Mamdani systems is the centroid [95]. The idea of this method is to get the centre of gravity of the aggregated fuzzy set. Let c_i be the centre of the membership function $\bar{U}_i(x)$, the centre of gravity can be computed by:

$$COG = \frac{\sum_i c_i \cdot \int_i \bar{U}_i(x)}{\sum_i \int_i \bar{U}_i(x)} \quad (2.2)$$

The set of methods chosen to simulate the logic inference, is known as *fuzzy inference engine*.

2.1.2 Takagi-Sugeno fuzzy systems

Takagi-Sugeno (TS) fuzzy systems [96] have been applied successfully in non-linear model based techniques [97] where the nonlinearity can be decomposed into multiple linear regions defined by each rule. In TS models, the system may be described by j rules by the following way:

Rule R_j :

IF x_1 is A_{x_1j}, \dots , and $x_n(k)$ is A_{x_nj} ,

THEN: $f_j = g_{0j} + g_{1j}x_1 + \dots + g_{nj}x_n$

being x_i, y_j for each rule, the inputs and outputs of the system respectively, and A_{x_ij} is the fuzzy set respective to $x_i(k)$ on the rule j , $g_i \in \mathbb{R}$, $f_j(k)$ is the output of the model respective to the operating region associated to that rule. The structure of antecedents describes fuzzy regions in the inputs space, and the one of consequents presents non-fuzzy functions of the model inputs.

These models may be formulated as an *adaptive neuro-fuzzy inference system* (ANFIS) [98]. In figure 2.3 an ANFIS is presented as an example with n input variables, one output variable and five layers. The first layer is composed of membership functions of

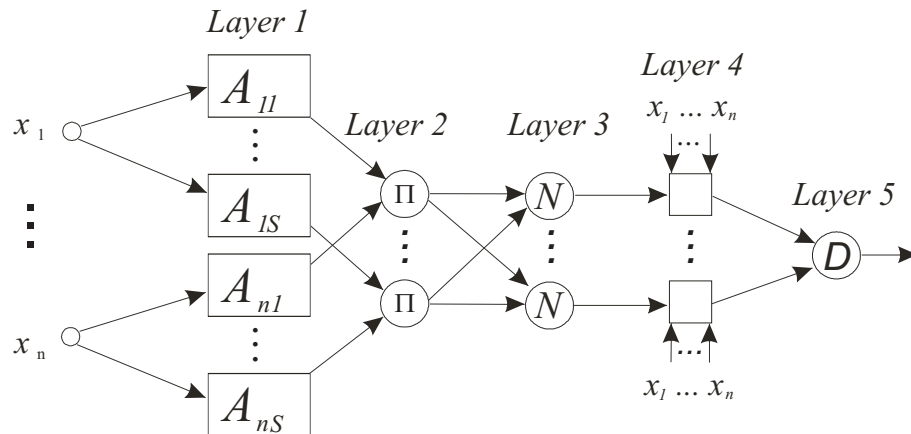


FIGURE 2.3: Fuzzy neural network [99]

each A_{ij} , defined by the membership degree

$$\mu_{A_{ij}} : x_i \in \mathbb{R} \mapsto \mu_{A_{ij}}(x_i) \in \mathbb{R} \quad (2.3)$$

The output of each node i is $\mu_{A_{ij}}(x_i)$, the membership degree of x_i . For the definition of these membership functions, some standard types are used, like gaussian membership

functions (see figure 2.2).

The second layer has nodes labelled with Π which implement fuzzy inference machine. Logical operation *AND* may be carried out by multiplication or minimum value for example, the output of each node j of this layer may be:

$$\omega_j(x) = \mu_{A_{1j}}(x_1) \cdot \mu_{A_{2j}}(x_2) \cdot \dots \cdot \mu_{A_{nj}}(x_n) \quad (2.4)$$

Or

$$\omega_j(x) = \min\{\mu_{A_{1j}}(x_1), \mu_{A_{2j}}(x_2), \dots, \mu_{A_{nj}}(x_n)\} \quad (2.5)$$

The third layer normalises the inference motor. The output of each node of this layer is:

$$a_j(x) = \frac{\omega_j(x)}{\sum_{j=1}^N \omega_j(x)} \quad (2.6)$$

Where N is the number of rules of the system. The fourth layer has adaptive nodes:

$$a_j(x) \cdot f_j(x) = a_j(x) \cdot (g_{0j} + g_{1j}x_1 + \dots + g_{nj}x_n) \quad (2.7)$$

Finally, the fifth layer is the defuzzification node. For TS systems, the output will be:

$$\sum_{j=1}^N a_j(x) \cdot f_j(x) = \frac{\sum_{j=1}^N \omega_j(x) \cdot f_j(x)}{\sum_{j=1}^N \omega_j(x)} \quad (2.8)$$

We will call $a_j(x)$ antecedent functions and $f_j(x)$ consequent functions. The output of the fuzzy complete model may be described by

$$f(x) = \sum_{j=1}^N a_j(x) (g_{0j} + g_{1j}x_1 + \dots + g_{nj}x_n) \quad (2.9)$$

2.2 Industrial standardisation

Although the advanced control, in each and every one of his strategies, has been widely used in many industrial applications, only some paradigms are defined by industry standards. Certainly the standard ISO/FDIS 15746-1 [100], about *Automation systems and integration - Integration of advanced process control and optimisation capabilities for*

manufacturing systems - in its Part 1: *Framework and functional model*, defines strategies as Model Predictive Control (and as examples: Dynamic Matrix Control, Generalised Predictive Control and Model Algorithmic Control), Fuzzy Control, Minimum Variance Control, Learning Control, Logic Control, Neural Network Control, Optimal Control and Statistical Process Control. However, there is a lack of standardisation about the way to implement practically each of them in a real plant. Only the fuzzy control has an industrial standard for deployment in industrial hardware platform: IEC61131 [11].

Fuzzy control is the theory of fuzzy logic applied to control engineering. Fuzzy control has emerged as a technology that can enhance the capabilities of industrial automation, and is suitable for level control tasks are generally performed in Programmable Logic Controllers (PLC).

The purpose of the standard IEC1131-7 is to provide manufacturers and users a common understanding well defined base, a means to integrate fuzzy control applications in the languages of Programmable Controllers according to Part 3 [7], as well as the ability to exchange fuzzy control portable programs between different programming systems. The standard defines the following terms:

Accumulation, aggregation, activation, conclusion, condition, crisp set, defuzzification, degree of membership, fuzzification, fuzzy control, fuzzy logic, fuzzy operator, fuzzy set, inference, linguistic rule, linguistic term, linguistic variable, membership function, singleton, subcondition, rule base, weighting factor.

The main part of the standard is the definition of *fuzzy control language* (FCL). The idea behind is to make the programmers be able to exchange fuzzy control projects between different manufacturers. The standard defines hierarchical levels of conformance:

- **Basic level.** With mandatory characteristics, data type and Function block definitions, etc that defines the minimum implementation.
- **Extension level.** Optional features for a more sophisticated implementation (more operators, defuzzification methods, etc.)
- **Open level.** Additional features related to the membership definition

It establishes the integration into the PLC using function blocks or programs previously defined in part 3[7]. The Standard calls also for the delivery of a defined data check list, in order to facilitate the transfer of applications among the different platforms. The figure 2.4 shows a Functional Block Diagram (FBD) [7] of FCL following the standard. In

it, basic level language functions blocks for fuzzification, operators and defuzzification are shown. To read more about this particular manufacturer FCL library, see [101]. In

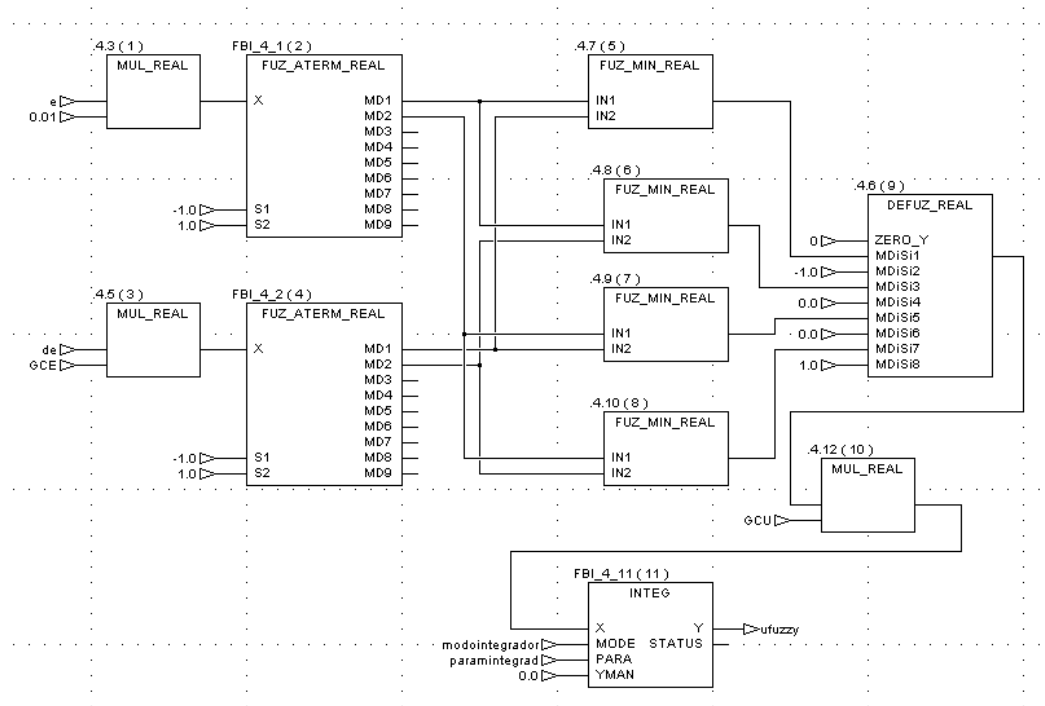


FIGURE 2.4: Example of FCL for a PLC [101]

Chapters 4 and 5, the standard IEC-61131-7 will be used for real applications.

2.3 Development of a FIS application for authentication gestures

As an application of use of fuzzy systems, we describe here a commercial implementation: authentication gestures [102, 103] Non-textual authentication methods differ in a number of ways from the classical username/password approach. Key among these, and the factor that drives the work presented in this section is that successful authentication follows not only from an exactly matching input, but from any one of the set of *sufficiently matching* inputs. While the textual password must exactly match the stored prototype, the non-textual input need only be sufficiently similar to the stored prototype since the exact match is exceedingly unlikely. The requirement for a proximity based match suggests that a fuzzy-based approach is appropriate. Since gestures cannot be repeated with precision, but can convey sufficient information to consider them *almost equal* to the stored prototype, fuzzy logic may be a suitable technique

to check for similarity. Fuzzy logic has been widely used in matching techniques [104]. Some techniques are based on *Fuzzy Transforms* [105],[106]. Others, on relative distance [107],[108],[109], on similarity measure [110],[111]. One of the typical field for application of fuzzy matching is the string and signature recognition, due to the ability of characters convey the same information using different graphical forms ([112],[113],[114],[115],[116]).

Given that the repeatability of a gesture lacks precision, but may contain sufficient elements to consider almost equal or not to the previously recorded, Fuzzy Logic may be a suitable technique to check the correspondence between them.

To recognise faces, an extraction of features can be performed using a biometric algorithm. Authentication based on a biometric factor is a widely used technique for mobile devices (i.e. [117]). Fuzzy logic is also an established method for matching those features [118]. A position of an image in a smart phone or tablet is composed of two coordinates (x_i, y_i) . The prototype will be composed of N points with the positions

$$P = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

A first prototype to be registered is built by the average of the position of several gestures given by the registration process. The *fuzzy matching engine* (FME) will make an index using the degree of membership of each pattern point to the prototype point.

There will be a fuzzy number defined for each prototype coordinate:

$$\tilde{P} = \{(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2), \dots, (\tilde{x}_N, \tilde{y}_N)\}$$

The fuzzy number \tilde{b} will be defined for the couple $\{b, d\}$ where b is the representative

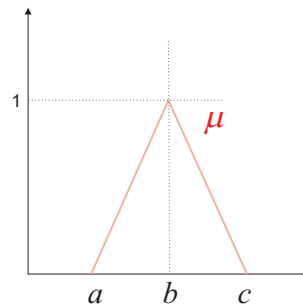


FIGURE 2.5: Triangular membership function

crisp number of \tilde{b} and d will be an adjusting parameter which defines the distance $d = |c - a|$ in the figure 2.5. In order to simplify the application, we will set it up with the same value for all the fuzzy numbers, calling it the *fuzziness* parameter.

Using a rule like: IF x_i IS \tilde{x}_i THEN $y = 1$, the degree of membership $\mu_{\tilde{x}_i}(x_i)$ of the crisp

number x_i to the fuzzy number \tilde{x}_i is obtained. Applying the rule to each coordinate gives a set of $\{\mu_{\tilde{x}_i}(x_i), \mu_{\tilde{y}_i}(y_i)\}$. Taking into account the sequence order and calculating each degree of membership, the expression

$$\mu = \frac{\sum_{i=1}^N \mu_{\tilde{x}_i}(x_i) \cdot \mu_{\tilde{y}_i}(y_i)}{N} \quad (2.10)$$

yields a matching index for the gesture and feature vector. A threshold value can then be used to establish whether the index value represents a match or not. This parameter is referred to as the *sensitivity*.

2.3.1 Result using the FME

Figures 2.6 and 2.7 show examples of matching (after adjusting fuzziness and sensitivity) using the matching index (2.10). Before applying the FME, the gesture is normalised to an image of the user's face in terms of orientation and scale so that comparisons can be made. For instance, the vector formed by joining the center of the eyes is a good reference. Figure 2.8 show how the gesture is matched with different orientations and sizes. The OpenCV library [119], was used for obtaining biometric markers

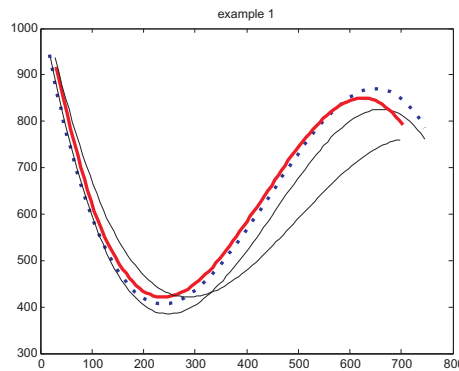


FIGURE 2.6: Example of matching. Dot line: prototype; Solid lines: not matched gestures; Thick solid line: matched gesture

on the face photographed by a camera connected or built-in to the device. The testing was performed on Windows/Android devices only. The touchStart/mouseDown events were used to capture the single-finger touch, the density of points captured depends on the device and also the speed at which the user swipes their finger. Therefore, an interpolation/extrapolation stage to produce 100 equidistant coordinates was used. Biometric markers [120] are used to produce a feature vector, which is given to the FME in order to identify the user. The testing results demonstrated so far that the FME can

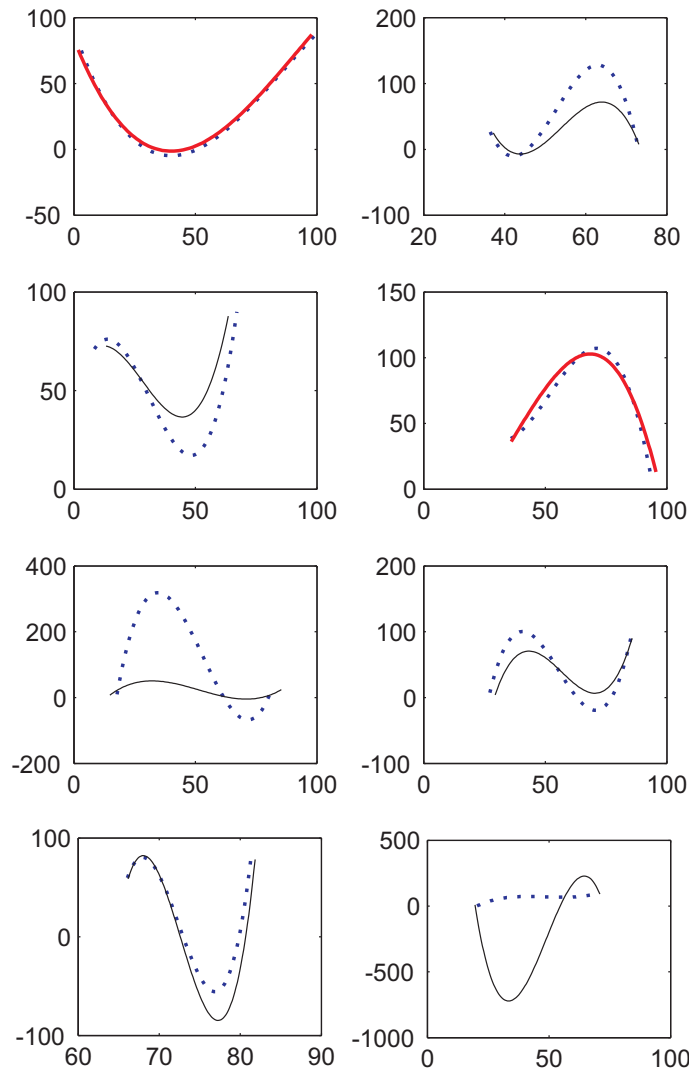


FIGURE 2.7: Examples of matching. Dot line: prototype; Solid line: not matched gesture; Thick solid line: matched gesture

be used for biometric/gesture authentication despite restrictions posed by the technical limitations of devices. Furthermore, a number of parameters such as the sensitivity (i.e. the threshold score for successful authentication) can be used to fine-tune a specific implementation. Figure 2.10 shows the distribution of scores for negative and positive gestures. *Positive* means a deliberate attempt to repeat a registered gesture while *negative* means a deliberate attempt to draw a *different* gesture. Generally, the negative gestures have a low score and the positive gestures have a high score. However, there is some overlap. This means that regardless the value that is chosen for sensitivity, false interpretations cannot be avoided. As it stands, the sensitivity is at 0.6, which leads to some rejected positives but does not allow any authenticated negatives.

- Scores for negative gestures were in the range 0 – 0.43 (mean=0.12, median=0.73)

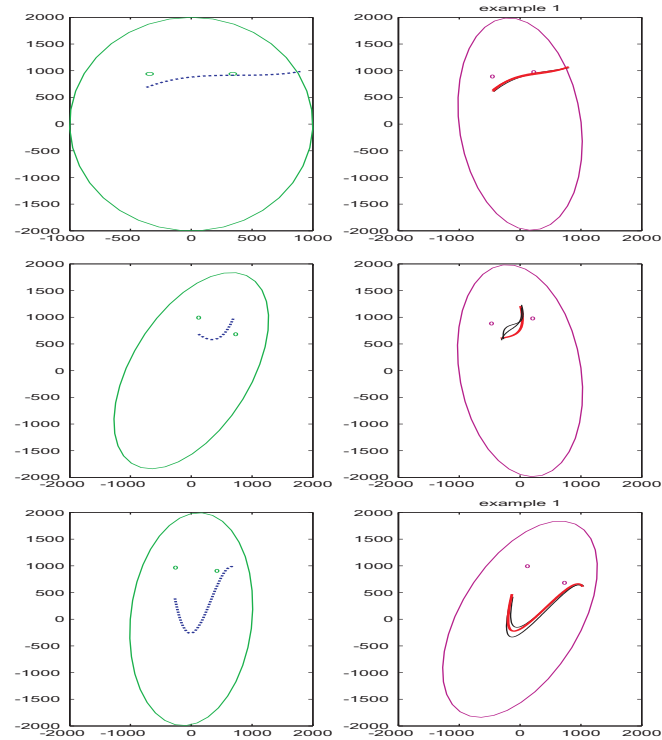


FIGURE 2.8: Example of matching after normalisation. Dot line: prototype; Solid line: not matched gesture; Thick solid line: matched gesture



FIGURE 2.9: Application working in a mobile device

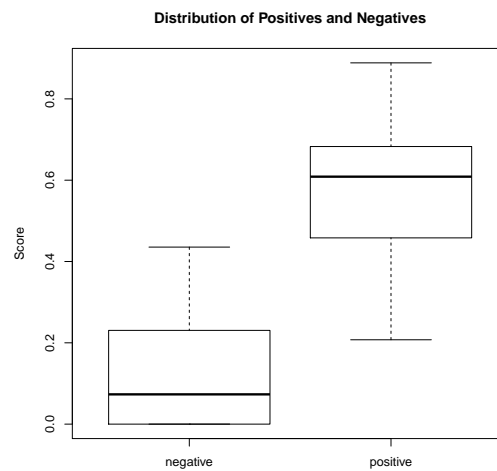


FIGURE 2.10: Distribution of positives and negatives

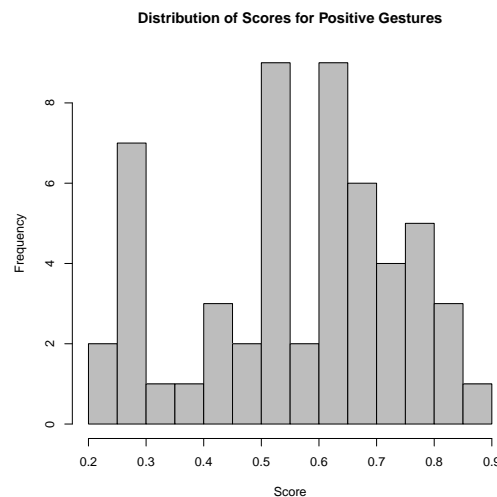


FIGURE 2.11: Distribution of scores for positives gestures

- Scores for positive gestures were in the range 0.21 – 0.89 (mean=0.56, median=0.61)

In this case, the threshold could safely be lowered to 0.5. However, there would still be no authenticated negatives but the number of rejected positives would have been reduced to 18% and authenticated positives would increase to 45%. Figures 2.11 and 2.12 show the distribution of scores for positive and negative gestures respectively. As expected, these are somewhat skewed to the left and right respectively. It should be noted that these statistics were compiled by manually examining gestures. The human operative would not know if a gesture was drawn from left to right or from right to left. However the system would always reject a gesture that was drawn in the opposite direction from its prototype. Consequently, some gestures labelled positive may, in fact,

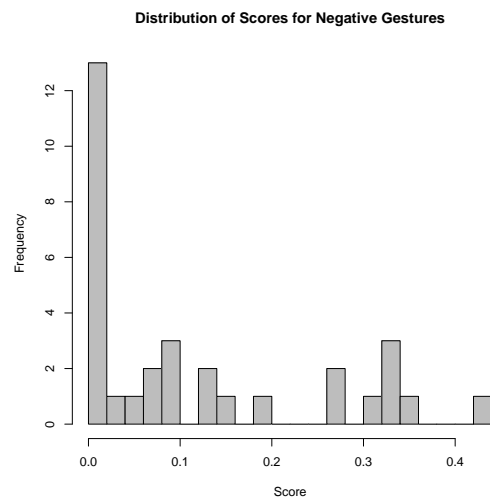


FIGURE 2.12: Distribution of scores for negatives gestures

have been negative. This may account for the unexpectedly high prevalence of positive gestures with a low score (the spike to the left on figure 2.11). Therefore, the system may have performed somewhat better than these numbers suggest.

2.4 Conclusion to the chapter

This chapter provides an overview of fuzzy inference systems. The industrial implementation through the FCL for PLC, shows that the use of fuzzy logic in the industrial world is well established, unlike other advanced control techniques. The standard provides manufacturers and users a common understanding well defined base, a means to integrate fuzzy control applications in the languages of Programmable Controllers according to the PLC standard languages, as well as the ability to exchange fuzzy control portable programs between different programming systems. A real application of the FIS has been described. It is a real example of commercial use of fuzzy techniques for authentication gestures in smart phones and tablets. The application is commercialised by the company Sensipass[®]. That application can be seen as a minor contribution for this thesis, which shows a practical implementation of a FIS for embedded systems.

Chapter 3

Fuzzy modelling techniques and applications

The use of models for prediction, simulation and control of systems is very common in engineering. There is vast literature regarding the modeling of dynamic systems. Obtaining a system model is not always an easy process. In many real cases, system complexity due to the number of variables in play, the random nature of the process, ignorance of the physics of the system, etc., makes it virtually impossible to obtain equations that reflect the behavior of the real system. At other times, an accurate model will consist of a large number of equations, which make the model impractical for use in control applications. In many cases, fuzzy inference systems can be the solution to the problem of modeling. Given that the behavior of systems can be described by rules captured by the experts and described with human language, fuzzy logic is an appropriate tool to formulate that knowledge.

Despite fuzzy models being highly successful in industry, even today, many scientists are reluctant to use fuzzy systems for modeling or control. This may be due to confusion caused by its name. Paraphrasing Professor Zadeh, the father of fuzzy logic, *"There are many misconceptions about fuzzy logic. Fuzzy logic is not fuzzy. Like traditional logical systems and probability theory, fuzzy logic is precise. However, there is an important difference. In fuzzy logic, the objects of discourse are allowed to be much more general and much more complex than the objects of discourse in traditional logical systems and probability theory."*[121].

The typical questions of the classical system identification community is: Why do we use a *fuzzy inference system* (FIS) for approximating continuous functions instead of more classical techniques, like regression, for performing this task? [122]. An answer

may be that a FIS forms a collection of fuzzy rules which can be extracted from experts knowledge, or from common sense. Moreover, each rule can represent a local model that is easily interpretable and analysable. This local type of representation enables improved approximation accuracy to be obtained [123]. One might object that Piece Wise Affine (PWA) models [124] already exist for that reason. However the transition from one region containing a linear model, to another region in which another linear model prevails, is naturally smooth and not sharp, as is the case with the PWA systems. This makes fuzzy systems more suitable than PWA models for describing nonlinear systems. The goal of fuzzy modeling is to obtain a set of rules that describe the dynamics of the system through experimental data. FIS are generic functional approximators, i.e., given a certain level of error, you can find a FIS that approximates any function with less than that fixed error. To do this, various techniques are used, some from the field of neural networks (NN) and also from other fields such as statistics, genetic algorithms, etc. This chapter will provide an introduction to the techniques of fuzzy modeling. In Section 3.1, the mathematical formulation of fuzzy models is presented, considering the problem of the choice of inputs and modeling error. In Section 3.2, the different groups will be most popular methods to determine the structure and parametrisation of these models. Finally, in section 3.3, several real applications made under this thesis will be presented.

3.1 Fuzzy modelling

Most real applications require mathematical models as functional assignments, corresponding actual inputs to outputs, where the aim is to approximate a function $y = f(x)$ in a limited area (compact) of input space $x = (x_1, x_2, \dots, x_n)$. In this section, the functional representation of fuzzy models will be described. As we mention in 2.1, there are two kind of Fuzzy Inference Systems, called Mamdani and Takagi-Sugeno fuzzy systems. Given a Mamdani system by rules of the form:

Rule R_j :

IF x_1 is A_{1j} AND x_2 is A_{2j} , AND, ..., AND x_n is A_{nj} ,

THEN: y_j is B_j

Where A_{ij} is a membership function (MF) of the input x_i and B_j the consequent of the output y_j . Using the product for the AND operator, the minimum for the implication, the union for the aggregation and $\mu_{ij}(x)$ being the degree of membership of the input x_i

to the fuzzy set A_{ij} and $v_j(y)$ represents the degree of membership of the output y_j to B_j .

To obtain the output, we can use the centroid defuzzifier:

$$y = \frac{\int_Y \mu(y) \cdot y \cdot dy}{\int_Y \mu(y) \cdot dy} \quad (3.1)$$

Where

$$\mu(y) = \bigcup_j \left[\min \left\{ \left(\prod_{i=1}^n \mu_{ij}(x) \right), v_j(y) \right\} \right]$$

Once the rule base is set, the problem of approximation is reduced to find the parameters that define each $\mu_{ij}(x)$ and $v_j(y)$, which usually are chosen as prototypical functions.

Examples are:

- Triangular

$$m(x; a, b, c) = \max \left[\min \left(\frac{x-a}{b-a}, \frac{c-x}{c-b} \right), 0 \right]$$

Where a, b, c (with $a < b < c$) determine the x coordinates of the three corners of the triangular MF

- Trapezoidal

$$m(x; a, b, c, d) = \max \left[\min \left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c} \right), 0 \right]$$

$a < b \leq c < d$ determine the x coordinates of the four corners of the trapezoidal MF

- Gaussian

$$m(x; c, \sigma) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma} \right)^2}$$

Where c is the MFs centre and σ determines the MFs width

- Generalised bell

$$m(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}}$$

c determines the centre of the corresponding membership function; a is the half width; and $\frac{b}{2a}$ controls the slopes at the crossover points.

3.1.1 Takagi-Sugeno models

Many methods can be found in the literature for the identification of a fuzzy model. One of the most popular methods is the formulation of a Takagi-Sugeno Fuzzy system as a NN [98], also called Adaptive Neuro Fuzzy Inference System (ANFIS). One of the classical ways to model complex systems experimentally is by using artificial NN. Models based on NN[125] are relatively easy to design, they often impose initial assumptions (as a functional dependency), simulation responses are quick. It is true that operators and engineers have a good insight into the operation of complex process, achieved over many years of experience. However a problem often arises when this knowledge is incomplete and imprecise, making the formulation of accurate mathematical equations very difficult. In general, all the training methods used in artificial NN can be transferred to the field of fuzzy systems [98], so called NeuroFuzzy Systems, which is an attempt to combine the learning ability of NN to the handling of uncertain information of the fuzzy inference system (FIS). The rule basis of an FIS can be based on expert knowledge, however, there are methods based on techniques that do not require a priori information about its structure.

FIS used for modeling, can handle nonlinear processes well, and provide knowledge of the system that is impossible with the use of NN. The models based on ANFIS combine the advantage of adaptive neural networks (ANN), such as the ability to learn and adapt, and fuzzy logic, i.e. knowledge based on rules and management uncertainty and significance of knowledge. Unlike ANNs based systems, ANFIS can incorporate a priori knowledge in order to improve the model.

In the neurofuzzy model proposed by Takagi-Sugeno (TS)[96], the structure of antecedent describes fuzzy regions in the inputs space, and the one of consequent presents non-fuzzy functions of the model inputs. Recurrent Fuzzy Neural Networks (RFNN) have demonstrated better results at identifying all the dynamics of nonlinear systems. They are systems which have the same advantages as recurrent neural networks [126, 127]. RFNN are also named *Fuzzy Dynamical Systems* (see figure 3.1) and extend the application domain of FNN to temporal problems. Feedback enables dynamics to be captured and updated. If we use recurrent functions with NARMAX structure (*Non-linear Auto Regressive Moving Average with eXogenous input*), of the kind: $\hat{y}(k+1) = f(y(k), \dots, y(k-m), u(k), \dots, u(k-n))$, where u, y are the inputs and outputs of the system, each rule of the system may be described by

$R_j :$

IF $x_1(k)$ is F_{1j}, \dots , and $x_n(k)$ is F_{nj} ,

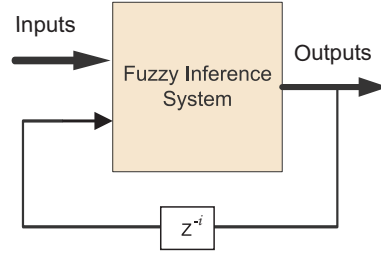


FIGURE 3.1: Dynamical Neurofuzzy System

THEN:

$$y_j(k) = a_j(z^{-1})y(k-1) + b_j(z^{-1})u(k-d) + \xi(k)$$

Where $a_j(z^{-1}) = a_{1j} + a_{2j}z^{-1} + \dots + a_{n_yj}z^{-(n_y-1)}$ and $b_j(z^{-1}) = b_{0j} + b_{1j}z^{-1} + b_{2j}z^{-2} + \dots + b_{n_uj}z^{-n_u}$

$X(k) = [x_1(k)x_2(k)\dots x_n(k)]^T$ is the inputs vector of the neurofuzzy system in the instant k , F_{ij} is the fuzzy set respective to $x_i(k)$ on the rule j , $y_j(k)$ is the output of the model respective to the operating region associated to the rule. If $\mu_{ij}(k)$ is the membership degree of $x_j(k)$ in the fuzzy set F_{ij} and the number of implications or rules is L , the Recurrent Fuzzy Neural Network (RFNN) complete model is described by

$$y(k) = \sum_{j=1}^L w_j(k) [a_j(z^{-1})y(k-1) + b_j(z^{-1})u(k-d)] + \xi(k) \quad (3.2)$$

Where

$$w_j(k) = \frac{\bar{\mu}_j(k)}{\sum_{j=1}^L \bar{\mu}_j(k)}, \quad \bar{\mu}_j(k) = \prod_{i=1}^n \mu_{ij}(k)$$

and $\xi(k)$ is a white noise sequence with zero mean.

Rewriting equation (3.2) as

$$\bar{a}(z^{-1})y(k) = \bar{b}(z^{-1})u(k-d) + \xi(k) \quad (3.3)$$

Where d is the delay and

$$\bar{a}(z^{-1}) = 1 - \bar{a}_1z^{-1} - \bar{a}_2z^{-2} - \dots - \bar{a}_{n_y}z^{-n_y} \quad (3.4)$$

$$\bar{b}(z^{-1}) = 1 - \bar{b}_1z^{-1} - \bar{b}_2z^{-2} - \dots - \bar{b}_{n_u}z^{-n_u} \quad (3.5)$$

$$\bar{a}_i = \sum_{j=1}^L w_j(k) a_{ij}z^{-i} \quad (3.6)$$

$$\bar{b}_i = \sum_{j=1}^L w_j(k) b_{ij}z^{-i} \quad (3.7)$$

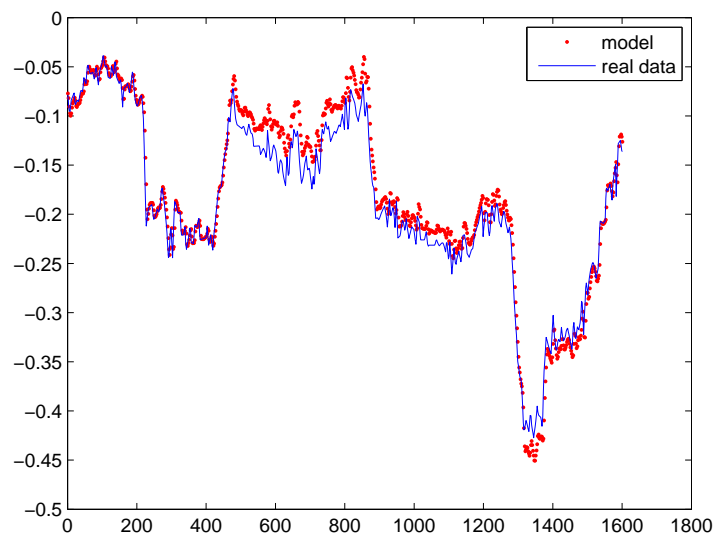
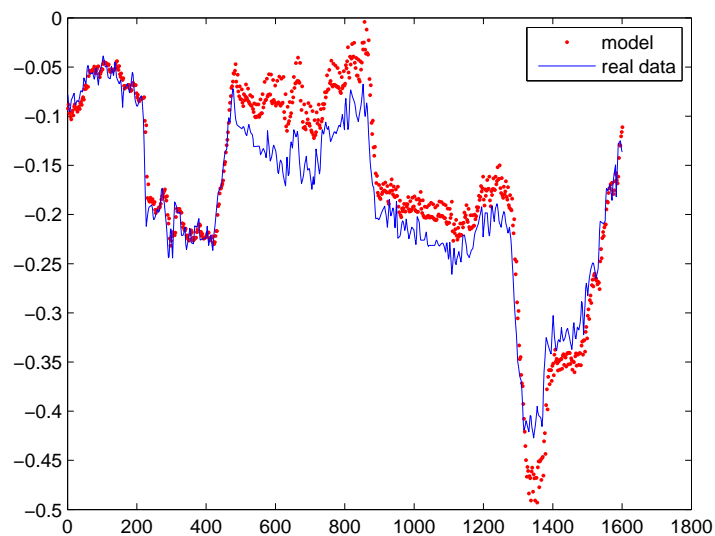
TS systems are computationally more efficient than Mamdani systems and work well with optimisation and adaptive techniques, which makes them very attractive in control problems, particularly for dynamic non linear systems [128].

3.1.2 Input selection

When establishing the structure ANFIS structure, one problem is to determine the membership functions. A large number will increase modelling accuracy, resulting in numerous management rules. This problem of the explosion of rules as linguistic terms, can be overcome with the application of clustering methods [129] seeking classification of data into subsets. In addition to the proper choice of rules and membership functions via a methodology, the choice of the input variables is important. In a simple application, consisting of a few variables, it is easier to choose the inputs, observing causality. When the system is composed of many variables, coupled with each other and hardly visible to the naked eye, a method can be to choose the maximum number of variables in the input FIS, and let any method from multivariate analysis decide the appropriate subsets, including the input.

On the other hand, in many applications, it is important to catch the dynamics of a system and predict it for a variable time horizon ahead. A model with previous samples of the output as inputs, may be useless if it's very sensitive to input errors, i.e., due to modelling error, a recurrent scheme may yield poor results if used in model-based control strategies with a large prediction horizon. For example, in figures 3.2, 3.3, 3.4 a comparison between a dynamical neurofuzzy model and the real data, in function of prediction horizon (one, three and five-step ahead) is presented. The mean absolute error $\overline{|e|}$ is used as a measurement of validation. While there is a huge literature on the modeling of fuzzy systems and its application to engineering, the literature reports few instances of a systematic methodology to obtain good models. Most of the contributions in this field refer to the methods for setting the parameters and the structure of fuzzy systems, but there is little reference made to the proper selection of inputs. In [130], a method based on testing models with different input selection is applied. The procedure for input variable selection is systematised as:

1. Evaluate the performance of the initial model with all candidate input variables in the model.
2. For each remaining input variable, evaluate the performance of the initial model with this variable temporarily removed.

FIGURE 3.2: One-step ahead model, $\overline{|e|} = 0.0142$ FIGURE 3.3: Three-step ahead model, $\overline{|e|} = 0.0284$

3. Permanently remove the variable associated with the best model performance obtained in step 2. Record the resultant reduced variable set and the associated model performance.
4. If there are still variables remaining in the model, go back to step 2 to eliminate another variable. Otherwise go to step 5.
5. Choose the best variable set from the sets recorded in step 3.

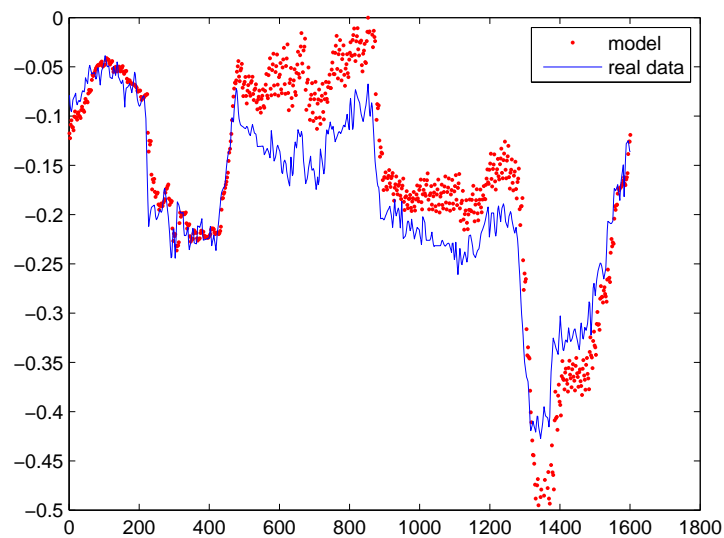


FIGURE 3.4: five-step ahead model, $\overline{|e|} = 0.0412$

The idea in [131] is to find the most relevant inputs by successively removing inputs from the initial input data set and checking whether the reduced data set is still consistent. Yen et al.[132] proposed using principal component analysis (PCA), a statistical analysis technique, to reduce the number of inputs for fuzzy models. Principal components are eigenvectors of the input variables' covariance matrix; the first principal component corresponds to a linear combination of the input variables that produces maximum variance in the input value (i.e., maximum input excitation) [133]. After a normalisation of the variables, the application of PCA will produce a new set of uncorrelated variables. However, it's important to select variables with any correlation with the output before PCA, because the selection is based on variability in the input value, not based on whether the input actually affects the output. An input variable with large variance may be completely unrelated to the output.

3.1.3 Fuzzy Time Series

The main objective of time series analysis is the construction of mathematical models based on known past instances of a variable in order to predict future values of the same. There are dynamic systems that exhibit repetitive dynamics in time or have a profile that appears regularly. When a system is complex enough to define the causal variables of dynamic effects, we have to resort to the study of time series to predict

future performances. For instance, Mackey-Glass equation [134] defined by

$$\dot{x} = \beta \frac{x(t - \tau)}{1 + x^n(t - \tau)} - \gamma x(t) \quad (3.8)$$

Where $\beta, \tau, \gamma, n \in \mathbb{R}^+$ Presents regularities as shown in figure 3.5 There are several

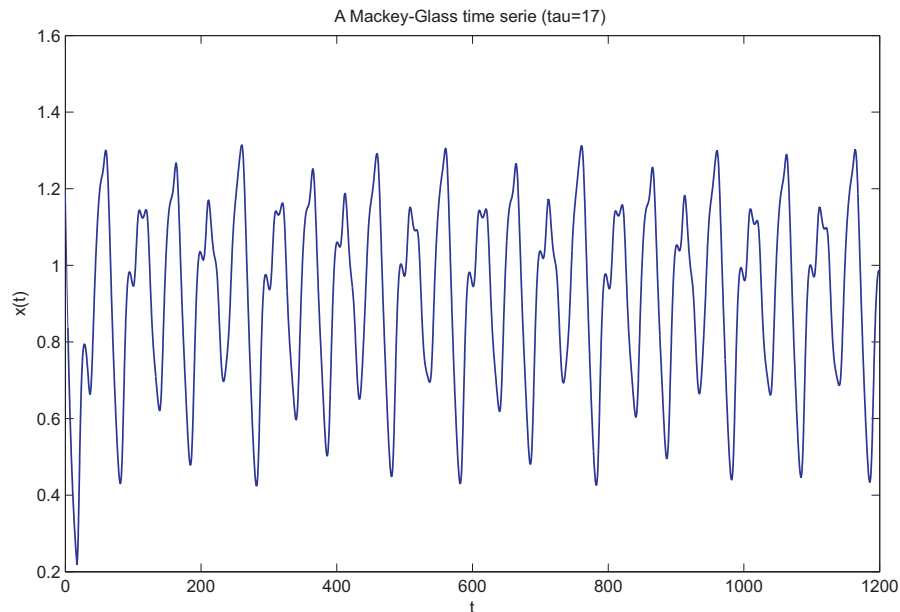


FIGURE 3.5: Mackey-Glass time serie equation. $\beta = 0.2, \gamma = 0.1, \tau = 17, x(0) = 1.2, n = 10$

methods to deal with the time series prediction [135]. Functional trend models are normally used, assuming that they are able to generate future values. This hypothesis is not suitable for forecasting medium or long range. In fact, time series regression analysis is widely used in economics, social science, biomedical data, etc. [136], [137], [138], in order to predict one step ahead. In [139] and [140] Fuzzy Time series were presented for first time. They can be used to deal with forecasting problems in which historical data are linguistic values. The procedure presented by Song and Chissom [139], [140], [141], and Santos and Arruda [142] presents a robust forecast even with non accurate data. Also S.M. Chen has improved song's method [143], [144], in [145] a comparison between them can be seen. However, these methods are not suitable for medium-long term predictions. Bocklisch and Päßler [146] propose a method based on the evolution of the membership functions following a course of time. Their models are suitable for short, medium and long range forecasts.

3.2 Review of methods to build and train fuzzy systems

There are numerous algorithms applicable to ANFIS learning to update the parameters of the layers. Notably, for example, Backpropagation(BP) [147] much used in NN. There are other methods combined with BP, such as hybrid combination of least-squares and backpropagation, [98], [148]. One of the advantages of fuzzy systems over NN is the addition of rules. These rules are given by the expert knowledge and observation of the system by the engineer. As discussed above, one of the most important properties of the FIS is its ability to approximate nonlinear functions with bounded approximation error. Improving accuracy causes an increase in the number of rules. It must be considered that a system based too many rules, is impractical as well as causing a loss of understanding of the system. The multivariable extension case is direct and in that case, the number of rules related to the defined number of MF's for each of the variables:

$$N = \prod_{k=1}^n n_k \quad (3.9)$$

The use of a clustering method, can avoid the rule explosion, getting the natural clusters between input and output variables. In this section, some popular and widely used techniques to obtain the structure and train the FIS to fit the real data are presented.

3.2.1 Clustering methods

Initially, in order to obtain the membership functions, it is useful to use a clustering method. Many algorithms exist for clustering analysis [149],[150], [151], [152]. One fast, one-pass algorithm for estimating the number of clusters and the cluster centers in a set of data, is the Subtractive Clustering (SC) [129]. This technique, like any clustering method is used to obtain the appropriate linguistic variables. The SC method is a modification of another: Mountain Method. In this algorithm, each point is assigned the potential

$$P_i = \sum_{j=1}^N e^{-\alpha \|z_i - z_j\|^2} \quad (3.10)$$

called mountain cluster, where $\alpha = 4/r_a^2$ y $r_a > 0$ defines the neighborhood radius for each cluster (is chosen depending on the desired resolution for groups). Let P_1^* the greatest potential, belonging to the point z_1^* chosen as the center of the cluster. Then,

for each point z_i reduced potential is calculated

$$P_i \leftarrow P_i - P_1^* e^{-\beta \|z_i - z_1^*\|^2} \quad (3.11)$$

The algorithm is as follows:

```

IF  $P_k^* > \varepsilon^u P_1^*$ 
  accept  $z_k^*$  as a new cluster and continue;
ELSE
  IF  $P_k^* < \varepsilon^d P_1^*$ 
    reject  $z_k^*$  and exit;
  ELSE
    let  $d_{min}$  be the minimum distance between  $z_k^*$ 
    and all centres found;
    IF  $\frac{d_{min}}{r_a} + \frac{P_k^*}{P_1^*} \geq 1$ 
      Accept  $z_k^*$  as the following cluster and continue;
    ELSE
      Reject  $z_k^*$  and assign potential 0;
      Select the point with highest potential
      as a new  $z_k^*$ ;
      Repeat the test;
    ENDIF
  ENDIF
ENDIF
ENDIF
ENDIF

```

Where ε^u specifies a threshold above which the point is selected as the center, and ε^d specifies a threshold below which the point is rejected. Typically $\varepsilon^u = 0.5$ and $\varepsilon^d = 0.15$. The radius for the reduction potential should be a degree higher than the radius of the neighborhood to prevent spaced clusters. Usually $r_b = 1.5 \cdot r_a$

3.2.2 Back propagation algorithms

Back-propagation algorithm was proposed in 1974 by Paul J. Werbos[147] and is the best known and popular feed-forward multi-layer NN method of learning. It is a iterative gradient algorithm. The basic principle is the error minimisation by using a gradient descent optimisation. Figure 3.6 illustrate the structure of a NN. Let ω_{ij}^k be the weight

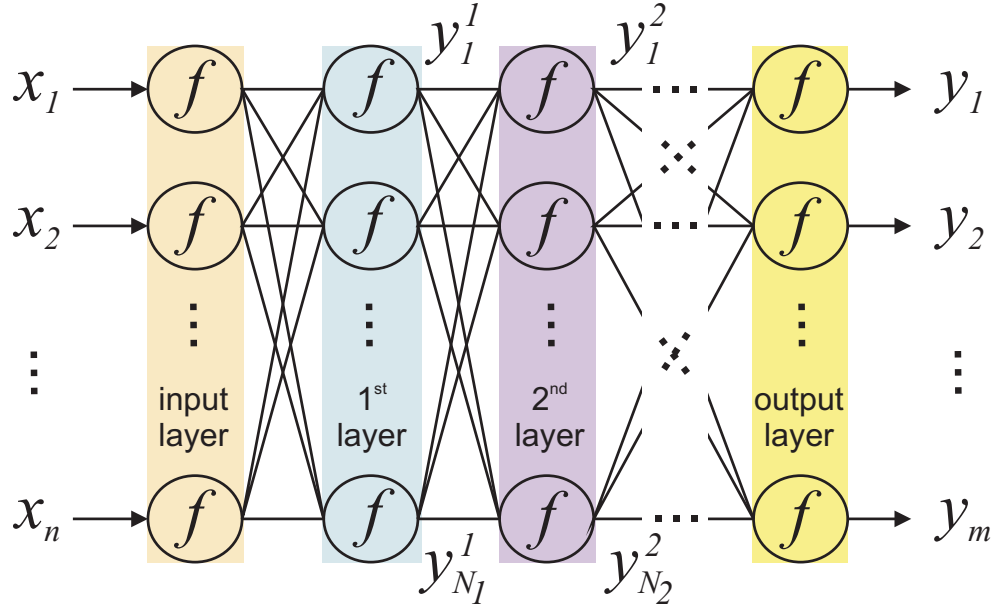


FIGURE 3.6: Structure of a multi-layer NN

of the i -th neuron in the k -th layer for the j -th input. The new weight will be given by:

$$\omega_{ij}^k(t+1) = \omega_{ij}^k(t) + 2\eta \delta_i^k(t) x_j^k(t) \quad (3.12)$$

where

$$\delta_i^k(t) = \varepsilon_i^k(t) f'(s_i^k(t)) \quad (3.13)$$

and

$$\varepsilon_i^k(t) = \begin{cases} \hat{y}_i^L(t) - y_i^L(t) & k = L \\ \sum_{l=1}^{N_{k+1}} \delta_l^{k+1}(t) \omega_{li}^{k+1}(t) & k = 1, \dots, L-1 \end{cases} \quad (3.14)$$

\hat{y}_i^L is the desired output value, being the output given by the activation function f :

$$y_i^k(t) = f(s_i^k(t)) \quad (3.15)$$

$$s_i^k(t) = \sum_{j=0}^{N_{k-1}} \omega_{ij}^k(t) x_j^k(t) \quad (3.16)$$

The value $\eta > 0$ is known as the *learning rate*, N_k is the number of neurons in the layer k . The algorithm can be expressed as follows:

Initialise weight to small values and set $\eta > 0$ to small value

WHILE (stopping criterion reached)

{

 Randomly select inputs and compute

 the output for each neuron using eq.3.15 and 3.16

}

```

    Compute error using eq.3.14
    Compute delta values using eq.3.13
    Update weight according to eq.3.12
}

```

The evolution of the algorithm will obtain a set of weights which minimise:

$$\sum_{i=1}^{N_L} (\hat{y}_i^L - y_i^L)^2 \quad (3.17)$$

To avoid *overfitting* [153], use a training set (on which the weights are adjusted) and a different set for prediction error evaluation. The backpropagation algorithm was the major breakthrough in the field of research of NN. However the algorithm is too slow for practical applications. To accelerate the convergence of the algorithm, many improvements have been made, for example: [154–159].

3.2.3 Evolutionary algorithms

Evolutionary algorithms (EA) are search and optimisation methods inspired by the principles of biological evolution solutions. In them a set of entities that represent possible solutions remains, which are mixed, and compete with each other, so that the fittest are able to prevail over time, evolving into increasingly better solutions. They are mainly used in problems with large spaces of variables and nonlinear relation among them, where other methods are not able to find solutions in a reasonable time. The entities that represent solutions to the problem are called individuals or chromosomes, and all of these together are called the population. Individuals are modified by genetic operators, mainly crossing over, the mixture consisting of the information of two or more individuals; mutation, which is a random change in individuals; and selection, namely the choice of individuals who survive and form the next generation. Because individuals that represent the most appropriate solutions are more likely to survive, the population is gradually improving. Evolutionary algorithms and evolutionary computation, are considered a branch of artificial intelligence. Genetic Algorithms (GA) are the common name used for EA for many people. A single population genetic algorithm works following the scheme of the figure 3.7. The selection determines, which individuals are chosen for recombination and how many offspring each selected individual produces. Recombination produces new individuals in combining the information contained in the parents. After recombination every offspring undergoes mutation. Offspring variables are mutated by small perturbations (size of the mutation step), with low probability. Genetic algorithms have a common basic outline. They process the whole set of solutions simultaneously. Individuals are named each potential solution to the problem. The algorithm works with the set of all individuals in the population. The composition of the population is modified along the iterations of the algorithm, called generations. From generation

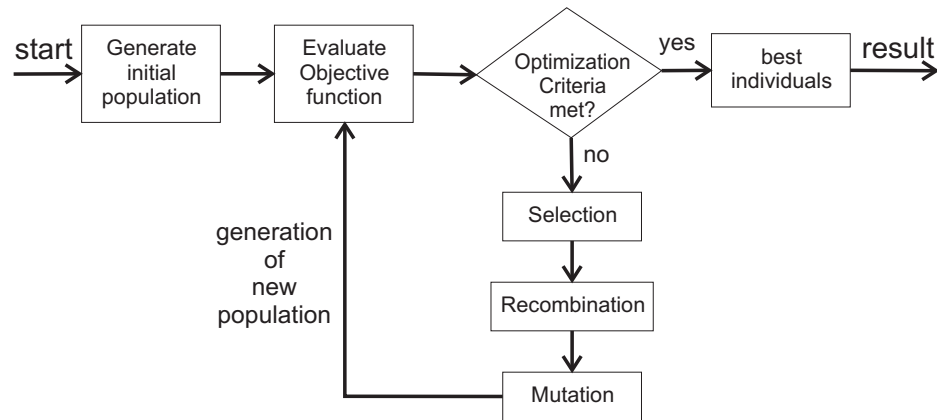


FIGURE 3.7: Structure of a single population genetic algorithm

to generation, in addition to varying the number of copies of a single individual in the population, new individuals may also be generated through transformation operations on individuals of the previous population. These operations are known as genetic operators. Each generation includes a selection process, which gives the best individuals more likely to remain in the population and participate in the reproduction operations. The *best guys* are those that minimise the algorithm adaptation function. It is essential for the work of an evolutionary algorithm that this process of selection has a random component, so that individuals with low adaptation also have opportunities to survive, although the probability is less. It is this random component which gives evolutionary algorithms ability to escape local optima and explore different areas of the search space. Variants best known are:

- Genetic algorithms (GA) [160]: use a binary or integer representation.
- Evolution Programs [161]: individuals are any data structure of fixed size.
- Real Coding Genetic Algorithm ([162–164]); is evolved a population of real numbers that encode the possible solutions of a numerical problem.
- Evolutionary Programming ([165, 166]); it evolves a population of "programs" to solve a problem in general. Evolving programs can take different forms, but the most usual is a tree. In any case, these data structures of variable size, ie, not all individuals have the same size. This type of EA presents a fundamental difference from the rest: do not seek the solution to a particular instance of a problem, but a strategy to solve any instance of this problem.

The use of GA for fuzzy systems, can not only generate the set of parameters required to train the fuzzy system but also its structure. For example, taking Mackey-Glass time series equation (3.8), an use a set of binary values to select which previous samples can be chosen as inputs,

Thus, in the following string: 101100101, the samples $x_{k-1}, x_{k-3}, x_{k-6}, x_{k-7}, x_{k-9}$ will be chosen as inputs for the FIS. GA can use the binary strings as individuals to evaluate an objective function, based on the model error between a FIS generated using that particular input structure and the series. In figure 3.8 a Fuzzy time series is compared with the Mackey-Glass equation. After the application of GA, the input set of the FIS is $x_{k-4}, x_{k-7}, x_{k-17}, x_{k-19}, x_{k-20}$, being \hat{x}_k the fuzzy time serie output, the Root Mean Squared Error (RMSE) one step ahead is:

$$RMSE = \sqrt{\sum_i^N \frac{(x_i - \hat{x}_i)^2}{N}} = 3.6131 \cdot 10^{-4} \quad (3.18)$$

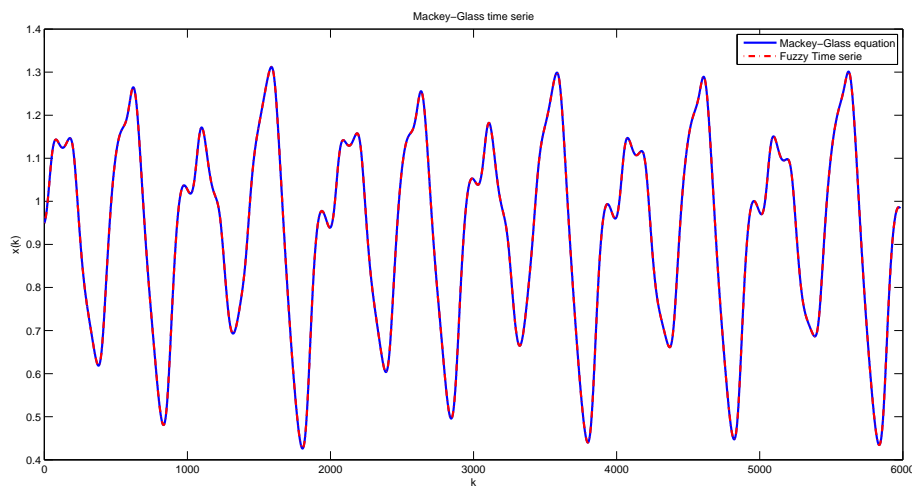


FIGURE 3.8: Mackey-Glass Fuzzy Time Serie

3.3 Development and validation of fuzzy models in real applications

In this section, three real applications carried out during the research work of the thesis are presented. The first is an application of modelling using Takagi-Sugeno models, based on linear models, which were determined from experimental data. The second one is also a TS model but differs by changing the input space for the subspace of the principal components (Principal Component Analysis). The third case study is an application of fuzzy time series for electrical demand prediction in a building.

3.3.1 Autoclave for food sterilization

Here an example of a fuzzy model being applied to a steam autoclave for the sterilization of food is given. This unit is located in Spain, in the IIM-CSIC (*Instituto de Investigaciones Marinas-Consejo Superior de Investigaciones Científicas*). The purpose of sterilization is to eliminate health risks by the thermal destruction of microorganisms, and to achieve product stability over long storage periods. The treatment is carried out at an elevated temperature ($> 100^{\circ}\text{C}$) so that



FIGURE 3.9: Autoclave for food sterilization (IIM-CSIC, Vigo, SPAIN)

the process is relatively short and, thus, having a minimal effect on factors of quality and nutrient retention. The unit uses a stream of saturated steam to heat the enclosed product. It has three inputs (steam, air and water) and two outputs (draining and purging). The sterilisation process is carried out in three stages: venting, heating and cooling. In the first stage, saturated steam passes through the vessel in order to evacuate the air from the system. Once the pressure into the vessel is equal to the saturated steam, the second stage starts. During the heating cycle, a fine control strategy is needed in order to track the reference and reject disturbances. After a predefined time, the third stage starts, cooling with water and controlling the pressure by injection of air. [167] In order to identify the process in the heating stage, a series of steps, of opening of the steam valve sequentially, have been done, for experimentally obtaining the temperature inside. Given the same increase in step, different temperature increases are obtained, as Figure 3.10 shows.

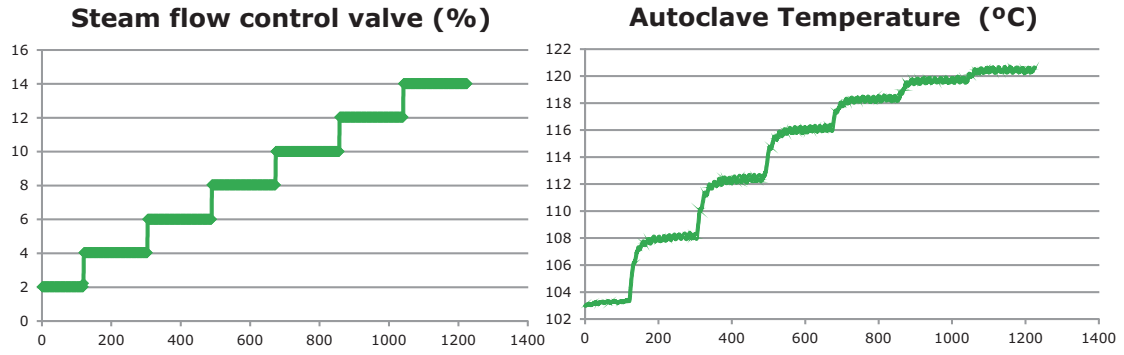


FIGURE 3.10: Autoclave experimental open loop test

The system can be described by a set of ordinary differential equations derived from mass and energy balances. The following assumptions are made in the derivation of the model equations of this unit:

- The steam, air and vapor-air mixture are considered ideal gases.
- The unit is heated homogeneously, i.e. the temperature is the same in all points of the autoclave.
- Liquid water and steam are considered in balance throughout the process.

Using Mass and Energy balance equations [168], is obtained:

$$[m_v(C_{pv} - R_v) + m_a(C_{pa} - R_a) + m_w C_{pw}] \frac{dT}{dt} = F_v^i [C_{pv}(T_v^i - T) + R_v T] + F_a^i [C_{pa}(T_a^i - T) + R_a T] + F_w^i [C_{pw}(T_w^i - T)] - F_p T [x_v R_v + x_a R_a] - \lambda \Psi - (Q_{rad} + Q_{conv}) - Q_{casc} - Q_{sol} \quad (3.19)$$

Where m is the mass, C_p the specific heat, F is the flow rate, x_y mass fraction of component y , R_y is the relationship between the universal gas constant (R) and the molecular weight of y , λ latent heat of water, Ψ water flow that is transferred between liquid and vapor phases, Q_r, Q_c, Q_h, Q_s , radiant, convection, through the casing, and through the solid heat, respectively, T the temperature. The subscripts v, a, w indicate vapor, air and water, respectively and the superscript i indicates "input". Equation 3.19 describes a nonlinear system and the nonlinear behaviour is evident in figure 3.10.

If we linearise at every equilibrium point, we will have different linear systems. In order to simplify the control problem and taking into account the nonlinear behaviour, a FIS can be made using the results of the experiment for identification. A set of rules formed by linear systems as a consequents can be set by the operating point variable values for the antecedents. For example, consider that $T(z)$ represents the autoclave temperature (in discrete time), and $U_s(z)$ represents the positions of the steam valve, and $G(z) = \frac{T(z)}{U(z)}$, we can formulate the rule-base of the FIS as:

$$\begin{aligned} \text{IF } T(z) \text{ is } 108^\circ\text{C THEN } G_1(z) &= \frac{0.09259z + 0.09259}{z - 0.9259} \\ \text{IF } T(z) \text{ is } 112.5^\circ\text{C THEN } G_2(z) &= \frac{0.07121z + 0.07121}{z - 0.9394} \\ \text{IF } T(z) \text{ is } 116^\circ\text{C THEN } G_3(z) &= \frac{0.06515z + 0.06515}{z - 0.9394} \\ \text{IF } T(z) \text{ is } 118^\circ\text{C THEN } G_4(z) &= \frac{0.05405z + 0.05405}{z - 0.9459} \\ \text{IF } T(z) \text{ is } 120.5^\circ\text{C THEN } G_5(z) &= \frac{0.03919z + 0.03919}{z - 0.9459} \\ \text{IF } T(z) \text{ is } 121.5^\circ\text{C THEN } G_6(z) &= \frac{0.00303z + 0.00303}{z - 0.9394} \end{aligned}$$

Where for the variable $T(z)$, a set of memberships have been chosen as shown in figure 3.11.

Figure 3.12 shows the validation of the Fuzzy model vs the real data.

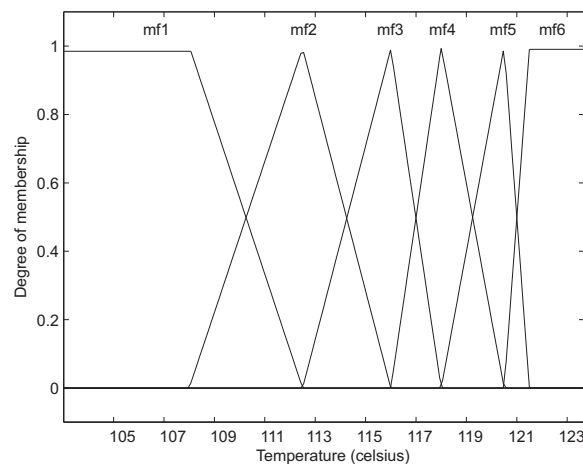


FIGURE 3.11: Membership functions of the temperature of Autoclave

3.3.2 Gas Mixing Chamber

As an application of PCA for Fuzzy modelling, a gas mixing chamber is proposed. The process is a part of Atlantic Copper Smelter facilities in Huelva (Spain), whose annual production is around three hundred thousand tons of copper [3]. This plant includes a Flash Furnace and four Pierce-Smith converters, two of them blowing simultaneously. The three gas streams generated in these processes are mixed in the mixing chamber and sent to three acid plants operating in parallel (see figure 3.14). It is very important to maintain the gas pressure in the mixing chamber at a desired value, always below ambient pressure in order to avoid gas losses to the atmosphere. That pressure depends on other variables of the production line and it is very difficult to get an accurate prediction of it. On one hand, the causes of the pressure oscillations are hard to detect. Moreover, since there are different control systems in the copper smelter and the acid plant, no clock synchronization is possible, so there are considerable uncertainties when

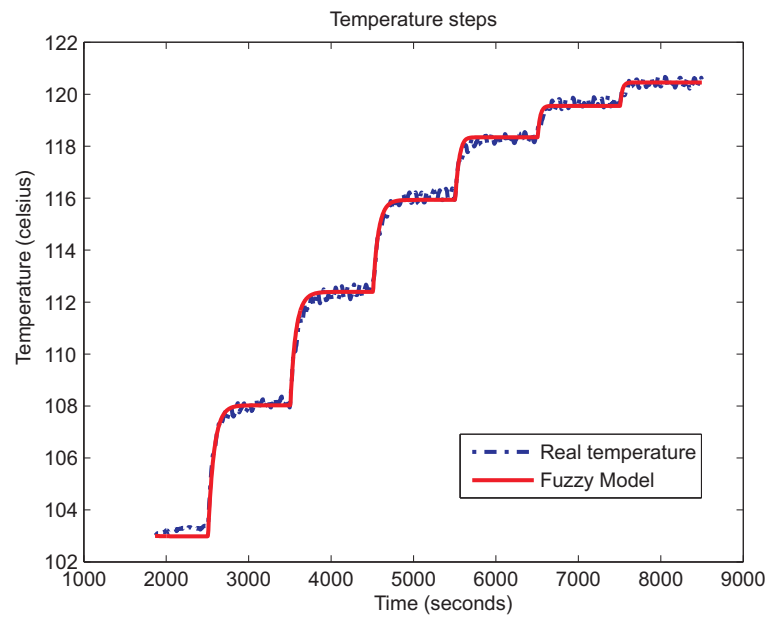


FIGURE 3.12: Real data and model output for the autoclave



FIGURE 3.13: General view of the copper smelter [3]

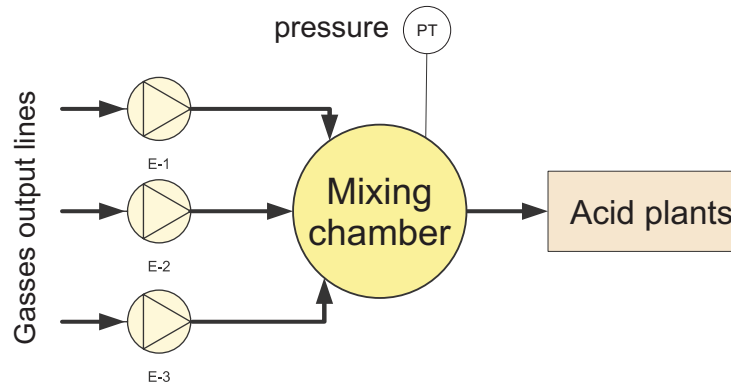


FIGURE 3.14: Gas mixing. Acid plants

trying to measure cause-effect delays. A suitable model for one step ahead prediction has been developed in [3]. Other models have been made in [169, 170]. All of them are prediction models, because the actual pressure value at time k $P_{MC}(k)$ is used to predict $P_{MC}(k+1)$. Taking into account that the converters operate on a batch mode, while those of the flash furnace and acid plants are continuous, extremely high disturbances both in flow and SO_2 concentration occur at the acid plants inlet due to the converters' operating schedule. The existing control strategy, based on independent single loop PID controllers, is not able to cope with those disturbances [3]. Advanced control schemes should be applied, but it would be interesting to have a model suitable for simulation. In this case, it is difficult to derive a precise mathematical model, based on first principles. Besides, the computation of the solution of models obtained through this methodology may require a large computational effort making them useless for real time tasks, such as control or optimisation. Neurofuzzy modeling, which permits an easy way to derive successful models, is a good alternative which can be employed to overcome such limitations [171–174].

After a preliminary study based on some experiments with steps on the variables, the evolution of the pressure in the mixing chamber (P_{MC}), is influenced by others that are divided into two groups: control signals and disturbances. In table 3.1 a brief description of the considered variables is given, whereas in figure 3.15 a scheme depicting each of them is presented.

Figure 3.17 presents the scheme followed to obtain the neurofuzzy model. The principal component analysis obtains new set of variables which are linear combination of the original input variables. Let $X = [x_1, x_2, \dots, x_n]$ a set of variables. If the principal components are y_1, y_2, \dots, y_p We can write

$$y_i = a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \quad (3.20)$$

and if $Y = [y_1, y_2, \dots, y_p]$, $Y = AX$ The idea is to find A to maximise the variance of Y , subject to $\langle a_{ik}, a_{jk} \rangle = 1$ if $i = j$ and $\langle a_{ik}, a_{jk} \rangle = 0$ if $i \neq j$. This analysis will be studied in more detail in chapter 6.

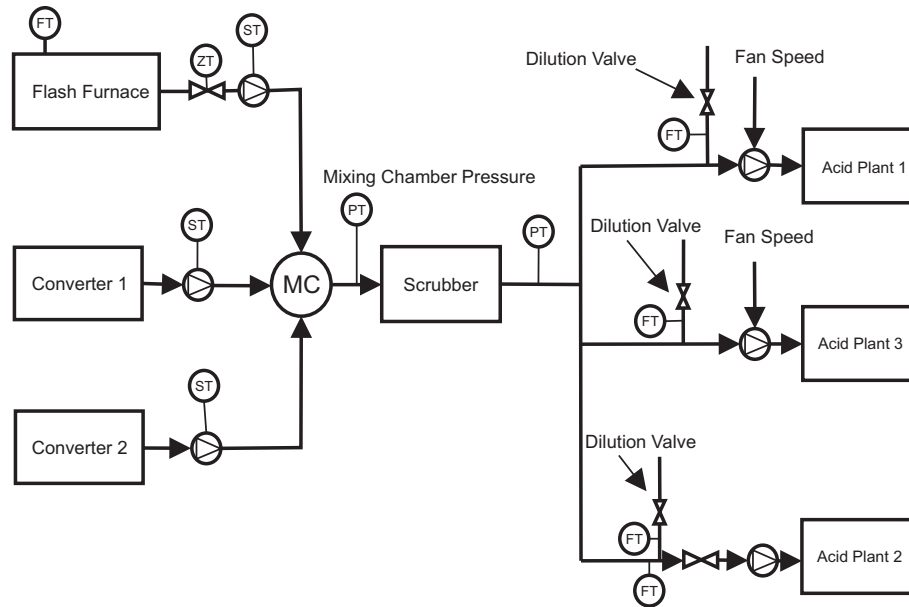


FIGURE 3.15: Process and manipulated variables

TABLE 3.1: List of variables

| <i>Description</i> | <i>Units</i> | <i>Type</i> |
|--------------------------------------|--------------|-------------|
| Pressure in mixing chamber | mbar | Output |
| Flow to plant 1 | kNm^3/h | Manipulated |
| Flow to plant 2 | kNm^3/h | Manipulated |
| Flow to plant 3 | kNm^3/h | Manipulated |
| Dilution flow to plant 1 | Nm^3/h | Manipulated |
| Dilution flow to plant 2 | Nm^3/h | Manipulated |
| Dilution flow to plant 3 | Nm^3/h | Manipulated |
| Reference for flash furnace feeding | Ton/h | Disturbance |
| Flow control valve for flash furnace | % | Disturbance |
| Fan speed in flash furnace | rpm | Disturbance |
| Reference for fan speed line 1 | rpm | Disturbance |
| Reference for fan speed line 2 | rpm | Disturbance |

The performance of the model can be seen in figure 3.16, where it is validated using a different real data set from the process.

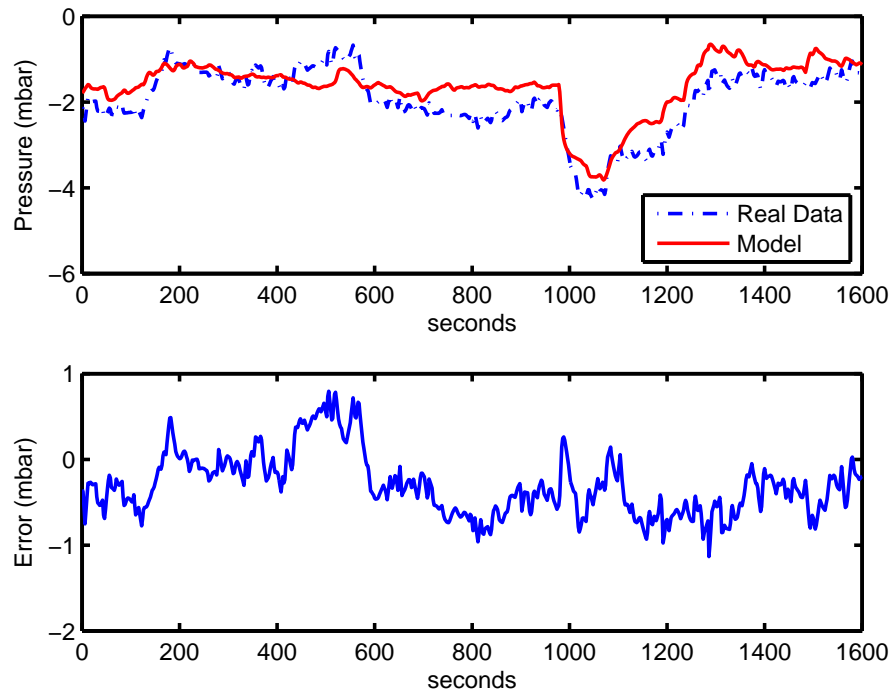


FIGURE 3.16: Validation of the model used in [170]

An improvement of that model is proposed using a major number of inputs, including squares of variables, to also provide a non linear dependence for each rule. A PCA has been used both in a model used in [170] and the one proposed here. In the first, the analysis is concerned with determining uncorrelated variables. In the second, further simplification is achieved in the FIS. The addition of inputs does not complicate the model when PCA is applied. Figure 3.17 presents

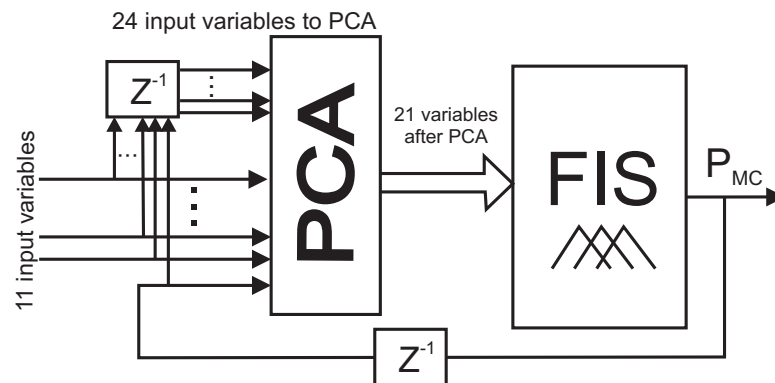


FIGURE 3.17: Model Scheme used in [170]

the scheme followed to obtain the neurofuzzy model. The inputs are the variables presented in table 3.1, including pressure in mixing chamber, their squares and the previous samples of all of them. To carry out the PCA, data have been used for approximately three hours of operation, sampled every 2 seconds. The first 7 components involve 99% of the variability of the data. Using these new 7 uncorrelated variables as inputs to the fuzzy system, and the next sampling pressure as output, an ANFIS is designed, using Subtractive clustering technique [172]. The performance of the model can be seen in figure 3.18, where it is validated using a real data set from the process. It is important to note that $P_{MC}(t-1)$ is generated using an independent model output, that is, the model is not fed with measured data [29]. Looking at the figures, it is evident that the error does not grow indefinitely, this fact makes the models appropriate to simulate the process. In figure 3.16, the mean error is 0.4099 mbar, while for the new model proposed, the mean error is 0.1469 mbar, obtaining a significant improvement of 64%.

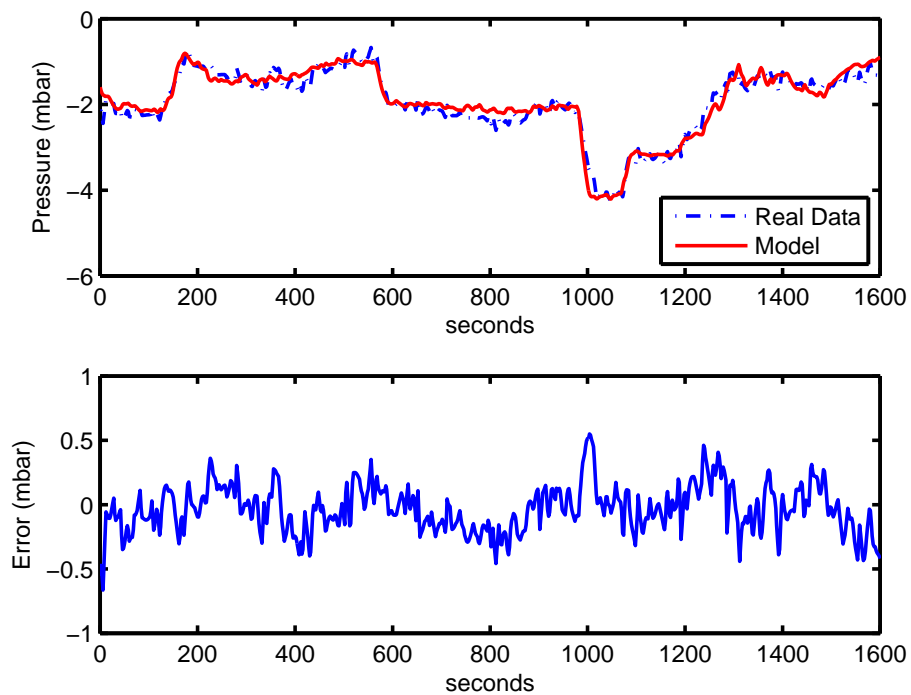


FIGURE 3.18: Validation of the proposed model used in [170]

3.4 Conclusion of the chapter

In this chapter, an overview of various fuzzy modeling techniques has been described. After a general description, the experience of several years of work with fuzzy models was presented. There have been two examples of modeling real industrial systems. The contributions to the

modeling of fuzzy systems, include using PCA to reduce the input space, allowing the modeling of complex systems. Accurate independent model outputs were obtained. In relation to the main objective of the thesis, this chapter helps to define an effective methodology for modeling non-linear and complex systems. Since MPC relies on accurate model predictions, the performance and accuracy of the independent model outputs demonstrated in this chapter, ensures that these models are suitable for predictive control.

Chapter 4

Fuzzy control systems: practical implementation

Fuzzy systems have predominantly been applied to control systems. In simple terms, a fuzzy control system executes control actions (outputs) using a rule base that evaluates some inputs (errors between the control variables and references). Fuzzy controllers (FC) can process simultaneously several variables from the system, hence they can also be considered as belonging to the class of multi-input–multi-output (MIMO) systems with interactions. The FC can be considered as a multi-input controller (eventually, a multi-output one, too), similar to linear or nonlinear state-feedback controllers [175]. Fig. 4.1 shows the classical diagram of a Mamdani fuzzy controller, where r is the reference input, y the controller output and d , the disturbance input.

Fuzzy control is not simply a technique of control. It is a general outline with room for multiple

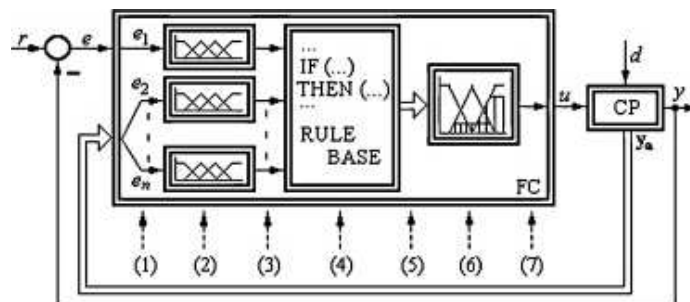


FIGURE 4.1: Basic fuzzy control system structure [175]. (1):crisp inputs, (2):fuzzi-fication module, (3):fuzzified inputs, (4):inference module, (5):fuzzy conclusions, (6):defuzzification module, (7):crisp output

control configurations where a fuzzy inference system is involved. In the majority of applica-tions a fuzzy controller is used for direct feedback control or on the low level in hierarchical

control system structure. However, it can also be used on the supervisory level, for example in adaptive control system structures. Nowadays fuzzy control is no longer only used to do model-free control. A fuzzy controller can be calculated from a fuzzy model obtained in terms of system identification techniques, and thus it can be regarded in the framework of model-based fuzzy control [175]. Reviewing the current literature, we could classify the fuzzy controllers in three groups: Direct fuzzy controllers, fuzzy model-based controllers and controllers with adaptive fuzzy parameters.

4.1 Direct fuzzy controllers

The direct fuzzy controller is an example of a Mamdani fuzzy system. In [176–179] the methodology to apply fuzzy controllers in a similar way to classical PID controller has been provided. Using a rule base, similar behaviour to that of classical linear controllers (P, PD, PI, PID) can be obtained. For example, in a classic discrete proportional controller, the output is a linear function of the input (error between the reference and the control variable)

$$U_k = K_p \cdot e_k \quad (4.1)$$

Where K_p is the controller constant (gain).



FIGURE 4.2: Fuzzy proportional direct controller

Using the structure shown in figure 4.2, with FIS being a representation of a fuzzy system, we can say that

$$U_k = f(GE \cdot e_k) \cdot GU \quad (4.2)$$

Where $f(x)$ is the input-output function determined by the set of rules. If the function is linear, such that $u_k = GE \cdot e_k$, we have that $U_k = GE \cdot e_k \cdot GU = K_p \cdot e_k$. A set of two membership functions can be established for the inputs and outputs to realise this relationship as figure 4.3 shows. The rule base for such controller would simply be:

IF e_k is P, THEN U_k is P

IF e_k is N, THEN U_k is N

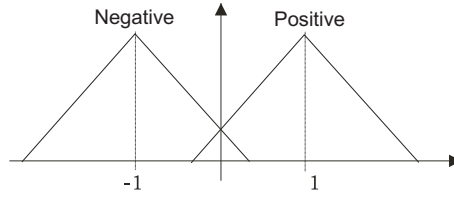


FIGURE 4.3: Memberships functions of the fuzzy PD direct controller

Setting $GE = 1$, GU will have the same meaning as the gain for a proportional controller. In a classical Proportional Derivative (PD) controller, the output can be given by

$$U_k = K_p \left(e_k + T_d \frac{e_k - e_{k-1}}{T_s} \right) \quad (4.3)$$

Where T_d is the derivative time and T_s the sample time.

In the fuzzy control scheme presented in figure 4.4, the inputs will be the error e_k and the error

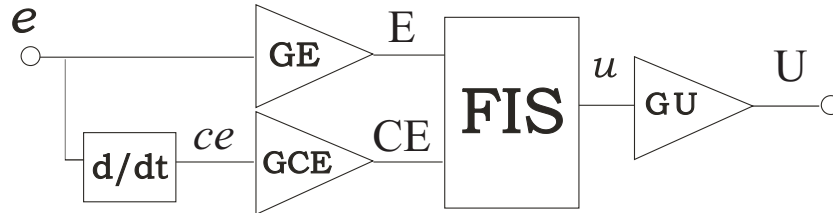


FIGURE 4.4: Fuzzy derivative direct controller

change Δe_k , where

$$\Delta e_k = \frac{e_k - e_{k-1}}{T_s} \quad (4.4)$$

Thus, the output is given by

$$U_k = f(GE \cdot e_k, GCE \cdot \Delta e_k) \cdot GU \quad (4.5)$$

As before, assuming a linear surface for the fuzzy system,

$$\begin{aligned} U_k &= (GE \cdot e_k + GCE \cdot \Delta e_k) \cdot GU \\ &= GE \cdot GU \cdot \left(e_k + \frac{GCE}{GE} \Delta e_k \right) \end{aligned}$$

Then, we have $T_d = \frac{GCE}{GE}$. The membership functions can be chosen, for example, to be equidistant, with a range of 0 to 1, as is shown in figure 4.5. And the rule base for this controller is:

- IF e_k is P AND Δe_k is P, THEN u_k is Big
- IF e_k is P AND Δe_k is N, THEN u_k is Medium
- IF e_k is N AND Δe_k is P, THEN u_k is Medium

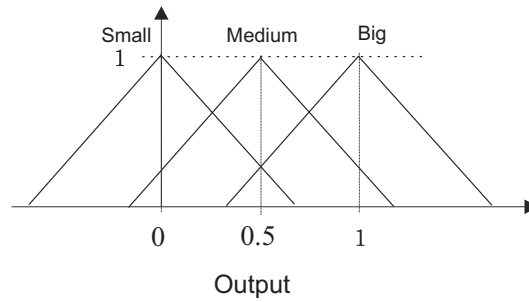


FIGURE 4.5: Memberships functions of the fuzzy PD direct controller

IF e_k is N AND Δe_k is N, THEN u_k is Small

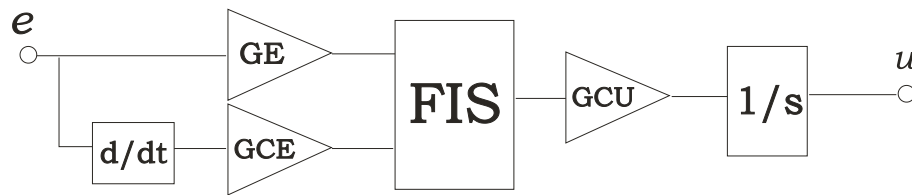


FIGURE 4.6: Fuzzy incremental direct controller

In order to include an integral action in a discrete controller, a summation of increments for classic controllers is given by $u_k = u_{k-1} + \Delta u(k)$, where

$$\Delta u(k) = K_p \left(e(k) - e(k-1) + \frac{1}{T_i} T_s \right)$$

Using the design of the figure 4.6, the output of the fuzzy controller will be $U(k) = \sum_i (\Delta u_i(k) GCU \cdot T_s)$.

A linear approximation for this controller is

$$U(k) = GCU \cdot \sum_{i=1}^k GE \cdot e(i) + GCE \cdot T_s$$

$$\begin{aligned} U_k &= \sum_{i=1}^k \left[GE \cdot e_i + GCE \cdot \frac{e_i - e_{i-1}}{T_s} \right] \cdot T_s \\ &= GCU \left[GE \sum_{i=1}^k e_i T_s + GCE \sum_{i=1}^k (e_i - e_{i-1}) \right] \\ &= GCE \cdot GCU \left[\frac{GE}{GCE} \sum_{i=1}^k e_i T_s + e_k \right] \end{aligned}$$

Comparing the constants with the classical PI,

$$\begin{aligned} K_p &= GCE \cdot GCU \\ \frac{1}{T_i} &= \frac{GE}{GCE} \end{aligned}$$

Two triangular membership functions can be chosen, for each input vertices -1 and 1 respectively and three for the output with vertices -1, 0 and 1 (Fig. 4.7)

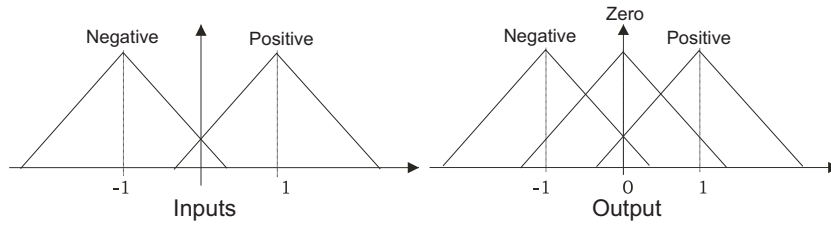


FIGURE 4.7: Memberships functions of the fuzzy incremental direct controller

The universe of the inputs can be chosen as the previous one and the output also with a range of -1 to 1. The basic rule base for this controller may be:

- IF e_k is P AND Δe_k is P, THEN Δu_k is Positive
- IF e_k is P AND Δe_k is N, THEN Δu_k is Zero
- IF e_k is N AND Δe_k is P, THEN Δu_k is Zero
- IF e_k is N AND Δe_k is N, THEN Δu_k is Negative

Due the output being an increment, this controller is known as fuzzy incremental controller (FIC) A fuzzy proportional derivative and incremental controller combines the structure of the above (FPD+FIC). Its basic arrangement is the one in Figure 4.8. Where the integral of the error is calculated:

$$ie_k = \sum_i (e_i T_s) \tag{4.6}$$

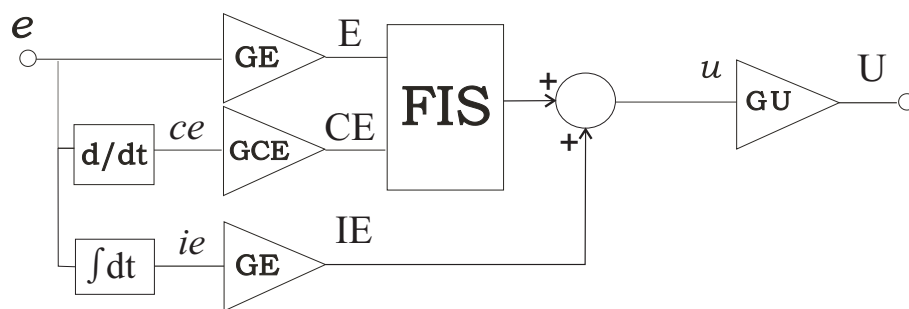


FIGURE 4.8: Fuzzy PD and incremental direct controller

And the output will be:

$$U(k) = [f(GE \cdot e_k, GCE \cdot \Delta e_k) + GIE \cdot ie_k] GU$$

the linear approximation is

$$\begin{aligned} U_k &= [GE \cdot e_k + GCE \cdot \Delta e_k + GIE \cdot ie_k] \cdot GU \\ &= GE \cdot GU \cdot \left[e_k + \frac{GCE}{GE} \Delta e_k + \frac{GIE}{GE} ie_k \right] \end{aligned}$$

Comparing the constants with the classical PID:

$$\begin{aligned} K_p &= GE \cdot GU \\ T_d &= \frac{GCE}{GE} \\ \frac{1}{T_i} &= \frac{GIE}{GE} \end{aligned}$$

The rule base for this controller is that of a FPD

4.1.1 Illustrative application: pneumatic levitation system

An implementation of a direct fuzzy controller for a pneumatic levitation plant has been done. As it is seen in section 4.1, the idea is to find the equivalence of the fuzzy controller's parameters and the parameters of a classical controller. Taking the error and its derivative as the inputs and incremental output (see fig.4.6), the controller is comparable to a classical PI controller. In fact, a way to adjust this controller would be to tune a PI first, and through its constants, calculate the fuzzy controller's parameters. The inputs are modulated by the constants GE and GCE (for error and error derivative respectively). The output will be the derivative of the manipulated variable that upon modulation with a constant GCU is integrated before acting. In a simple way, two triangular membership functions for each input vertices -1 and 1 respectively and three for output with vertices -1, 0 and 1 (Fig. 4.7) can be chosen.

The rule base for this type of controller may be that seen in section 4.1. To adjust the constants (GE, GCE, GCU), the constants of a PI (K_p, T_i) must be adjusted first and then, the following calculation is performed:

$$K_p = GCG \cdot GCU \quad \frac{1}{T_i} = \frac{GE}{GCE}$$

For instance, $GE = 0.1$ is chosen and GCE and GCU are calculated. Figure 4.9 shows the control surface for this controller. Compared to a conventional PI controller, the surface is nonlinear. A good use for this property is that it can treat nonlinear systems.

The system is shown in figure 4.10 [180]. The device consists of a centrifugal fan blower driven by an AC motor connected to a variable-speed drive. The drive speed-reference is sent from a commercial PLC, which receives the ball position measurement from a personal computer

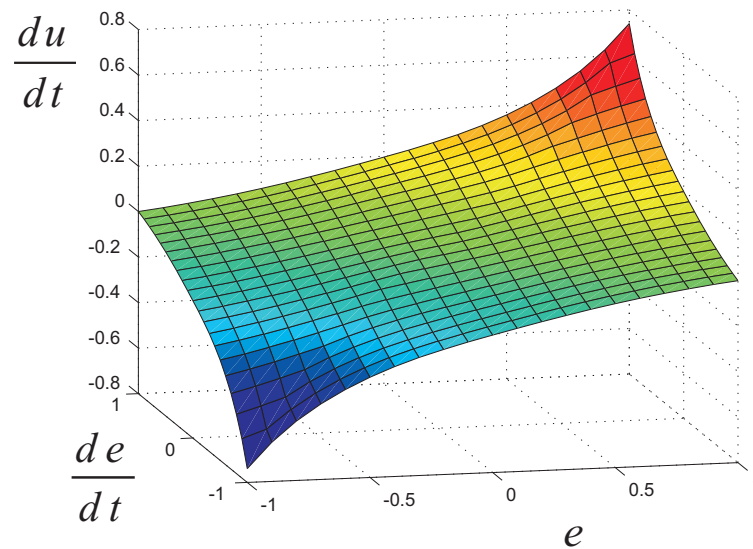


FIGURE 4.9: Control surface of a fuzzy incremental direct controller

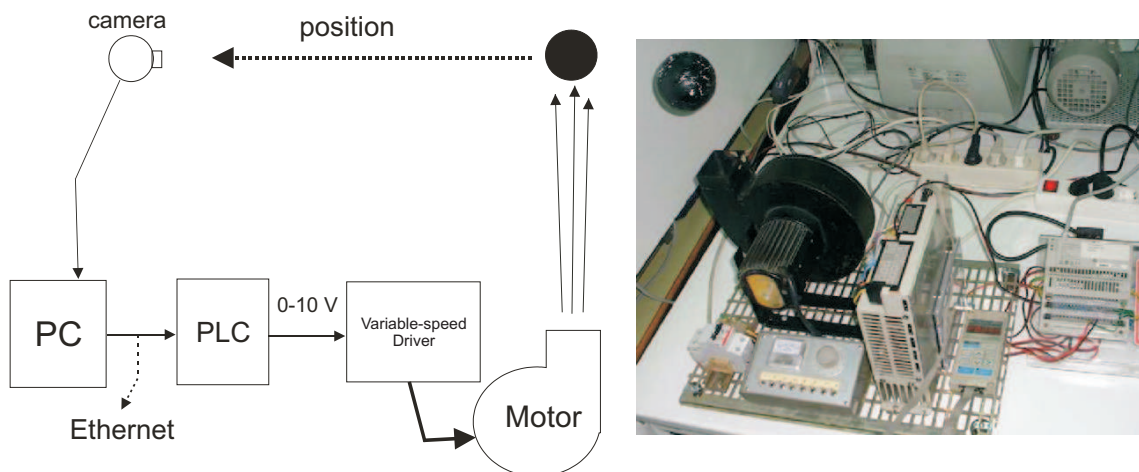


FIGURE 4.10: Pneumatic Levitation System [180]

linked to it through Ethernet using the MODBUS TCP/IP protocol. Finally, the ball position is computed by means of a camera connected to the PC, which has an appropriate image processing software available [181]. As noted in [180], there is a dependence of the air flow velocity, v and the distance to the fan output, h . Although a wide zone where the velocity can be considered to be proportional to h^{-1} exists, if h is large enough (in this application, about 50 cm) the flow velocity decreases drastically, making the system dynamics very complex. Moreover, taking into account that other effects (such as lateral and rotational motions, lack of sphericity, rugosity, etc...) have not been considered, it is easy to realise that the behaviour of this system is subject to a nonlinear dynamics. Presumably a linear controller shall have adequate performance, where the system shows a linear behavior. In order to compare, three control strategies have been implemented. The first controller is a classical PI, which provided a good trade-off between tracking and disturbance rejection as can be observed in fig. 4.11. A model-based H_∞

controller was also tried out (Fig. 4.12). Although a better behavior was achieved for tracking experiments, this controller provided poor performance for disturbance rejection. This fact may be explained taking into account the uncertainty of the system dominant mode, which is too oscillatory. The direct fuzzy control test can be seen in (Fig. 4.13). Despite having only a few rules, this controller has slightly improved performance over that attained by the PI controller. The advantage of applying a direct fuzzy controller in this case, is clearly seen in the table 4.1, where the *Integral of Time Absolute Error* (ITAE) index of the time responses are exposed.

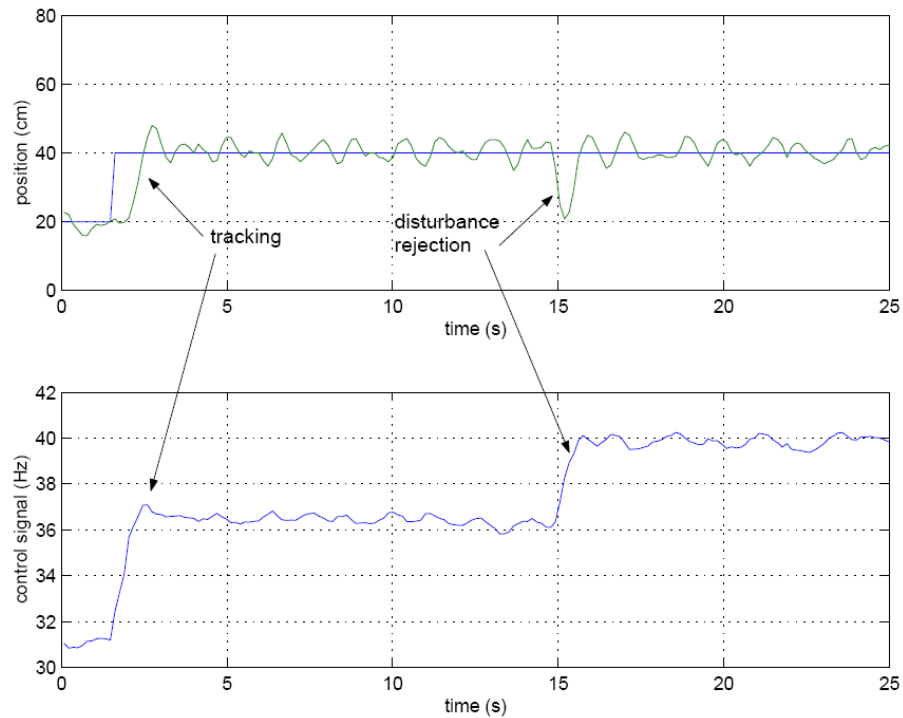


FIGURE 4.11: Experimental results with the PID controller

TABLE 4.1: ITAE index for each controller

| <i>Controller</i> | <i>Tracking</i> | <i>Disturbance rejection</i> |
|-------------------|-----------------|------------------------------|
| PID | 19.42 | 10.88 |
| H_{∞} | 17.54 | 16.12 |
| Fuzzy | 18.66 | 9.72 |

This controller has been implemented on a industrial PLC following the standard IEC61131-7 [11], (see 2.2). The code of the fuzzy block, according with this standard is:

```
FUNCTION_BLOCK fuzzy1
VAR_INPUT
    e REAL; (* RANGE(-1 .. 1) *)
    de REAL; (* RANGE(-1 .. 1) *)
```

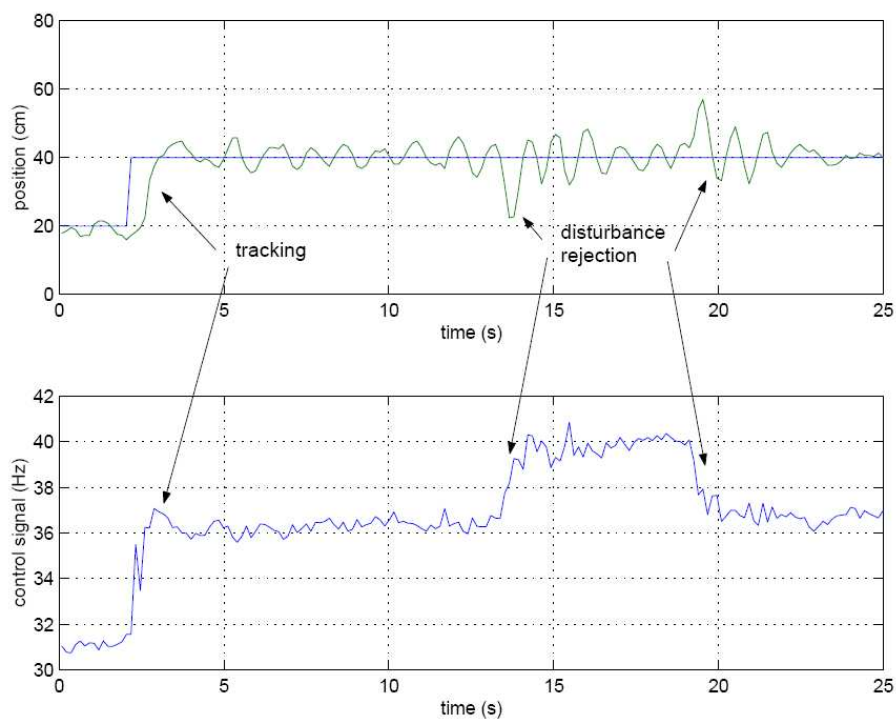


FIGURE 4.12: Experimental results with the H_∞ controller

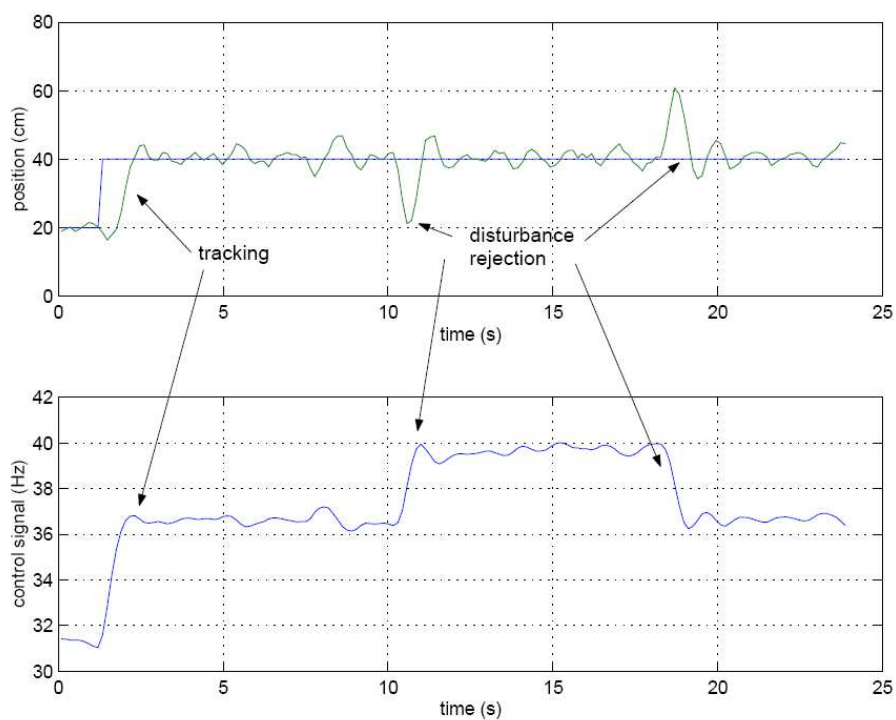


FIGURE 4.13: Experimental results with the fuzzy controller

```

END_VAR
VAR_OUTPUT
    T_0 REAL; (* RANGE(0.7460 .. 0.9928) *)
END_VAR
FUZZIFY e
    TERM e_m1 := (-1.0, 1) (1.0, 0);
    TERM e_m2 := (-1.0, 0) (1.0, 1);
END_FUZZIFY
FUZZIFY de
    TERM de_m1 := (-1.0, 1) (1.0, 0);
    TERM de_m2 := (-1.0, 0) (1.0, 1);
END_FUZZIFY
DEFUZZIFY du
    TERM du_m1 := -1.0 ;
    TERM du_m2 := 0.0 ;
    TERM du_m3 := 1.0 ;
    METHOD: MoM;
END_DEFUZZIFY
RULEBLOCK No1
    AND:MIN;
    ACCU:MAX;
    RULE 0: IF (e IS e_m1) AND (de IS de_m1) THEN (du IS du_m1);
    RULE 1: IF (e IS e_m1) AND (de IS de_m2) THEN (du IS du_m2);
    RULE 2: IF (e IS e_m2) AND (de IS de_m1) THEN (du IS du_m2);
    RULE 3: IF (e IS e_m2) AND (de IS de_m1) THEN (du IS du_m3);
END_RULEBLOCK
END_FUNCTION_BLOCK

```

Where the inputs e and de have to be previously multiplied by the constants GE and GCE respectively and the output du needs to be multiplied by GCU before being integrated.

4.1.2 Stability analysis of fuzzy control systems

Literature about studies of stability of fuzzy control systems abound, e.g. for direct controllers: [182–192], and for fuzzy model-based control systems: [193–204]. In [205] and [206] a design algorithm is proposed in order to obtain stable fuzzy controllers. Li-Xin Wang [207, 208], raised an interesting scheme based on adding a supervisory control for fuzzy controller. Figure 4.14 shows the scheme proposed. The control action is composed by $u = u_c + u_s$ with u_c being the fuzzy controller action and u_s the

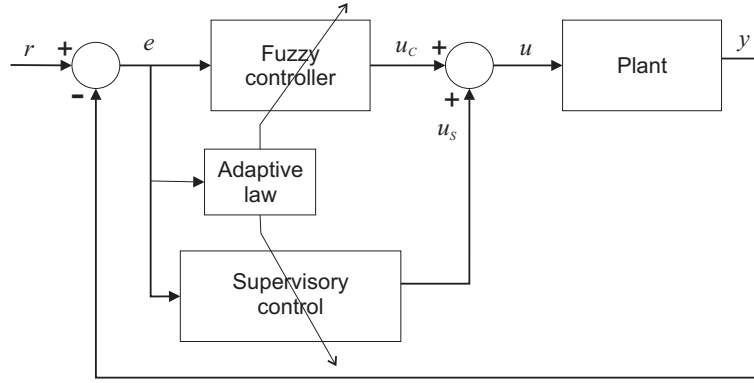


FIGURE 4.14: Scheme of adaptive supervisory control[208]

supervisory controller action. The idea of this control scheme is to compute a term u_s based on the structure of the controller and apply it to avoid violating stability limits established by a Lyapunov function. Within the limits of stability $u_s = 0$.

4.2 Fuzzy model-based control

One of the well known techniques that provided a procedure to design a controller from a TS fuzzy model is the Parallel Distributed Compensation (PDC). In [209], Kang and Sugeno design a way to control a system using a TS model. In [210–212] a procedure was developed, taking into account the stability of the system in the design. Using the state space formulation, the i th rules of a TS fuzzy model will be [213]:

IF $z_1(t)$ is M_{i1} and ... $z_p(t)$ is M_{ip} THEN

$$\begin{cases} \mathbf{x}(t+1) &= \mathbf{A}_i\mathbf{x}(t) + \mathbf{B}_i\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}_i\mathbf{x}(t) \end{cases}$$

Where $z_i(t)$ are the input premises of the system. They can be state variables or function of the inputs or disturbances. M_{ij} are fuzzy sets defined for the variables $z_j(t)$. The consequent is a linear system with outputs vector $\mathbf{y}(t)$ and state variables vector $\mathbf{x}(t)$

If for each rule we design a linear fuzzy controller such that

IF $z_1(t)$ is M_{i1} and ... $z_p(t)$ is M_{ip} THEN

$$\mathbf{u}(t) = -\mathbf{F}_i\mathbf{x}(t) \tag{4.7}$$

The global state vector will be

$$\mathbf{x}(t+1) = \sum_{i=1}^r a_i(\mathbf{z}(t)) \{ \mathbf{A}_i \mathbf{x}(t) + \mathbf{B}_i \mathbf{u}(t) \} \quad (4.8)$$

The overall fuzzy controller will be

$$\mathbf{u}(t) = - \frac{\sum_{i=1}^r w_i(\mathbf{z}(t)) \mathbf{F}_i \mathbf{x}(t)}{\sum_{i=1}^r w_i(\mathbf{z}(t))} = - \sum_{i=1}^r a_i(\mathbf{z}(t)) \mathbf{F}_i \mathbf{x}(t) \quad (4.9)$$

substituting 4.9 into 4.8,

$$\mathbf{x}(t+1) = \sum_{i=1}^r \sum_{j=1}^r a_i(\mathbf{z}(t)) a_j(\mathbf{z}(t)) \{ \{ \mathbf{A}_i - \mathbf{B}_i \mathbf{F}_j \} \mathbf{x}(t) \} \quad (4.10)$$

According to [214] and [210] we can state the following sufficient stability condition [210]:

Theorem 4.1. *The equilibrium of a fuzzy system is globally asymptotically stable if there exists a common positive definite matrix \mathbf{P} such that for all subsystem i ,*

$$\mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P} < 0 \quad (4.11)$$

Finding a common \mathbf{P} is a LMI problem. From equation 4.7 and theorem 4.1, the following theorem can be formulate [211, 215]:

Theorem 4.2. *The equilibrium of a fuzzy control system is globally asymptotically stable if there exists a common positive definite matrix \mathbf{P} such that the following two conditions are satisfied:*

$$\{ \mathbf{A}_i - \mathbf{B}_i \mathbf{F}_i \}^T \mathbf{P} \{ \mathbf{A}_i - \mathbf{B}_i \mathbf{F}_i \} - \mathbf{P} < 0, \quad i = 1, 2, \dots, r \quad (4.12)$$

$$\mathbf{G}_{ij}^T \mathbf{P} \mathbf{G}_{ij} - \mathbf{P} < 0, \quad i < j \leq r, s.t. \quad a_i \cap a_j \neq \emptyset \quad (4.13)$$

The control design problem is to select $\mathbf{F}_i (i = 1, 2, \dots, r)$ such that conditions 4.12 and 4.13 are satisfied.

As seen in Chapter 1.1, the current paradigm of model-based control is the predictive control. The use of fuzzy techniques in predictive control was proposed for the first time by [216]. In the literature there are two approaches to conceive using fuzzy inference systems on predictive control. One is based on the use of fuzzy optimisation to solve the problem of predictive control, an approach that transparently translates objectives

and constraints to predictive control by fuzzy multicriteria decision making [217–219]. Following this line, [220] a stable model-based fuzzy predictive control based on fuzzy dynamic programming is proposed. Other approach is the use of fuzzy systems as a prediction model for any MPC strategy, [221–225]. The use of fuzzy modeling for predictive control is justified by the complexity of the system to be modeled, either because the number of variables and their interactions, as the nonlinearities or hybrid nature [226]. In chapter 5 a FMPC application will be studied in more detail.

4.3 Controllers with adaptive fuzzy parameters

The FIS are often used to modify controller parameters, following an adaptive control scheme. In [227–230], a fuzzy system is used to adapt online the parameters of a PID. This strategy has also been used to adjust the MPC parameters setting [231–234]. For example, in [231], an evaporator process has been regulated by an MPC with a fuzzy parameter adaptive system. In that work the idea was to establish a set of rules changing the adjustments of the MPC parameters as a function of the degree of bound violation. That reactive change gives the system a better performance as the fuzzy system will provoke changes before the bounds are violated.

4.3.1 Air Separation Unit

Another application done for this thesis is an adjustment of the move suppression in a MPC for one Air Separation Unit. The cryogenic process is comprised of unit operations that compress, purify, and separate the air feed into the required gaseous and liquid oxygen, nitrogen, and argon product flows ¹[235]

The argon column gets its feed as an intermediate vapor stream from the low pressure column which is mostly returned to the low pressure column as liquid condensed by that crude oxygen stream. Clearly there is significant interaction (Fig.4.16) when almost any of the manipulated variables are adjusted or a disturbance affects one of the column controlled variables.¹

In this process, one of the important variables is the *mid point purity* (MPP). This is the gaseous oxygen plus argon product leaving the low pressure column. In a continuous operation process, it is important to keep this value as stable as possible.

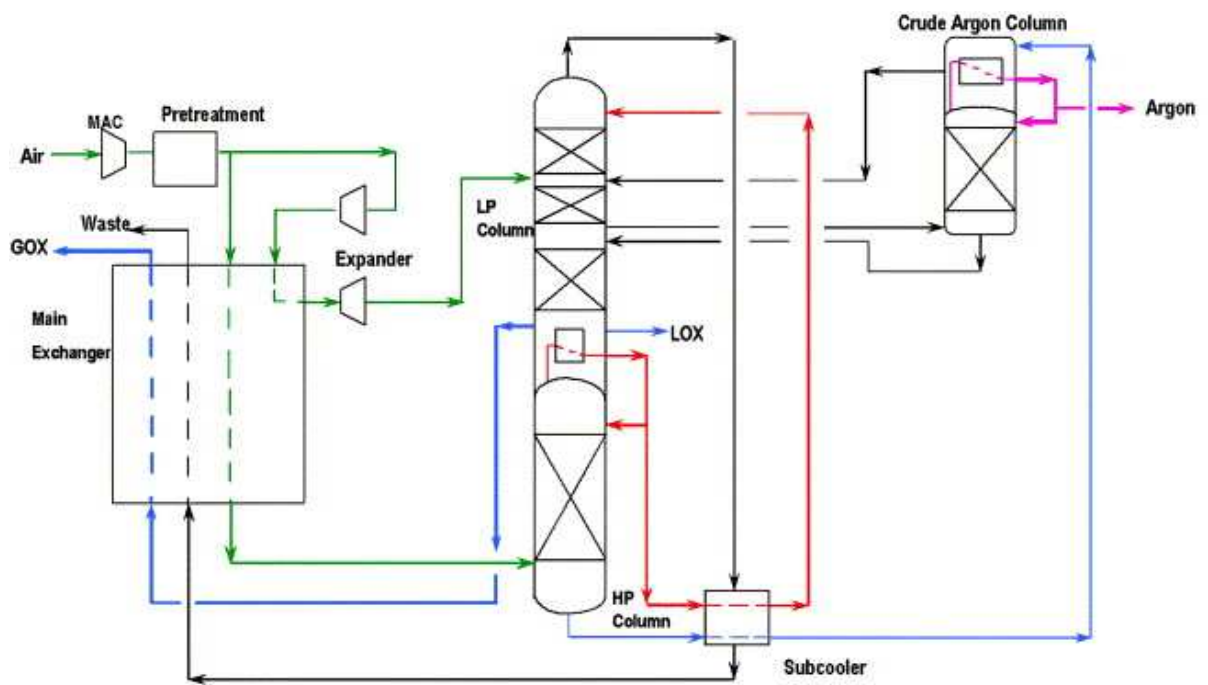


FIGURE 4.15: Simplified process for the cryogenic production of oxygen and argon ¹

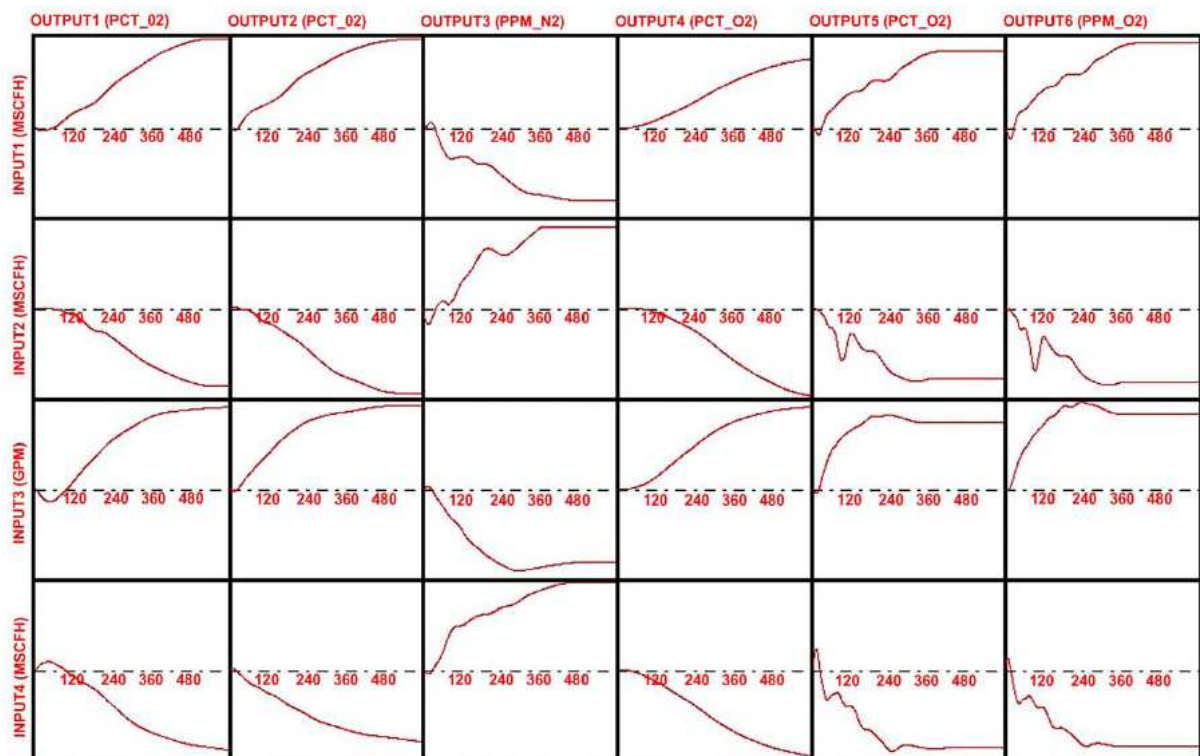


FIGURE 4.16: Portion of the dynamic matrix for a cryogenic air separation plant ¹

There are several disturbances that affect this variable: input air to the high pressure column and the reflux from the high to the low pressure column. When the molecular sieve changes, a disturbance affects the MPP. A fuzzy adaptive system has been designed to modulate the move suppression in the MPC of the MPP control depending on the molecular sieve operating stage, making the controller less aggressive during that disturbance. An air separation plant supplying gaseous products is either connected to a large pipeline system supplying multiple customers or it may be directly supplying product to a single customer. Regardless, the product must be supplied to the customer when it is needed, so the plant has to respond rapidly to the changing product demand. Given the long time to steady state, it is possible that the plant may rarely ever settle into a steady state condition. This requires that the control strategy handle both the dynamic and steady state effects in order to achieve efficient operation. At the same time, the energy intensive nature of cryogenic liquid production often requires changing production rates to take advantage of variable power pricing or supply chain demands.¹

A setting of a controller that allows a sufficiently aggressive action to changes in production can lead to a greater sensitivity to the disturbance caused by changes sieves. The designed fuzzy system takes into account the variation in demand, as measured by total oxygen flow out of the plant. When the system is experiencing changes in production, the rules cause a more aggressive driver, manipulating the main flow of air compressor that affects the MPP, to keep control variable at the set-point. Decreasing the variation in demand the fuzzy supervisor increases the move suppression, causing smoother control action, which will be not excited by the periodic disturbances created by changing sieves. The action of this adaptive system provides better performance of the MPC, achieving a reduction of about 10% in the standard deviation of the MPP regarding its set-point.

Figure 4.17 shows the performance of this control scheme (b). Comparing with a MPC with fix move suppression (a), the controller reacts smoothly with sieves changes (periodic disturbances in the incoming air), but with more aggressive changes with the oxygen demand. Although the change in the demand for oxygen is lower in the graph (a), an improved performance compared to MPP control with fuzzy supervisor (b) can be seen.

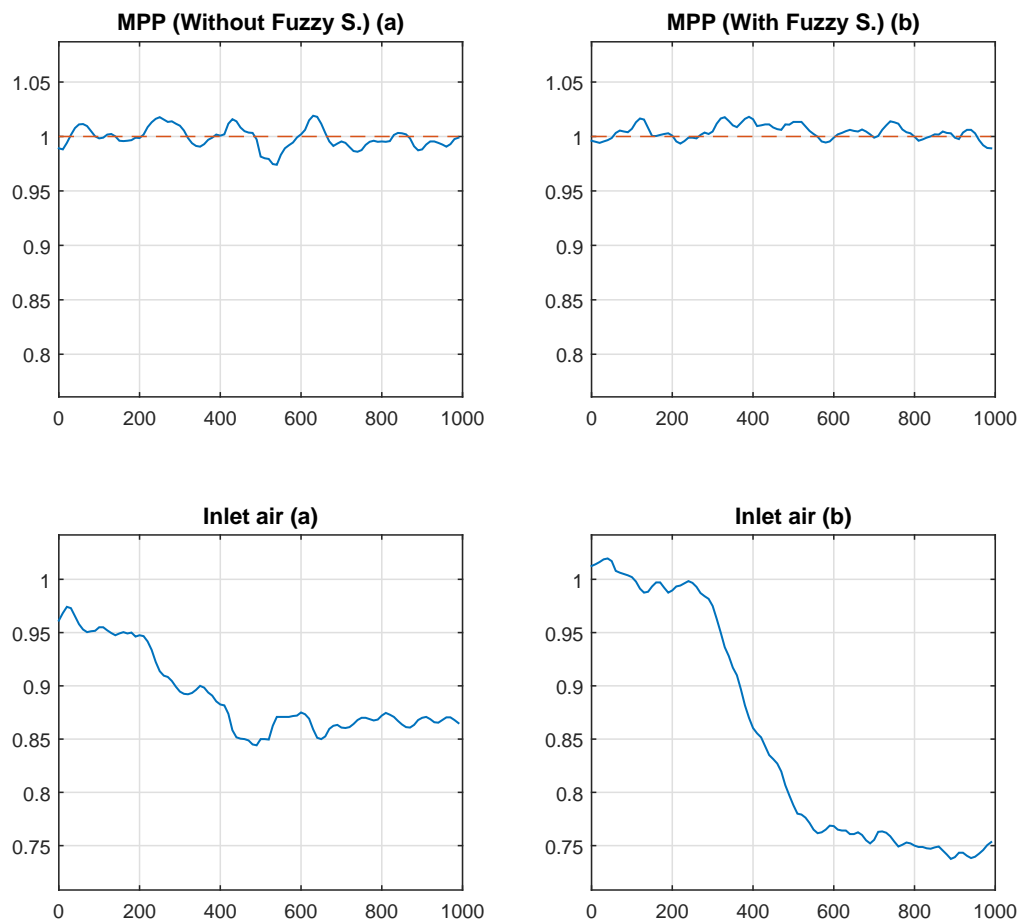


FIGURE 4.17: MPP control without (a) and with fuzzy supervisory system (b). Normalised variables

4.4 Conclusion of the chapter

In this chapter, the main fuzzy control techniques have been summarised. Fuzzy control is a large field of study where many engineers have designed different control strategies [175]. There are excellent books and articles related to fuzzy control Systems [175, 236], even surveys related to specific fields of applications [237, 238]. In addition to the summary, the major contribution of the chapter is the actual implementation of control strategies in real plants. A comparison of other drivers with a direct fuzzy controller in air levitation plant has been made and an application of a supervisory fuzzy control system in a real chemical plant in operation has been applied. In the next chapter we will focus on a specific application of fuzzy model-based predictive control.

Notes

¹Reprinted from [235] with permission from Elsevier

Chapter 5

Fuzzy MPC for an industrial autoclave

In this chapter, Fuzzy Model-based Predictive Control (FMPC) will be described in more detail. Fuzzy control techniques were introduced in chapter (4), and MPC was described in chapter 1. As it was seen, in FMPC a fuzzy model is used in order to design an optimum controller for a prediction horizon, in order to minimize tracking error and control action effort. Nowadays, as seen in chapter 1, Model Predictive Control is considered a well established technology in many fields, especially in industrial process applications. Its efficiency has been demonstrated over the last few years. In general, most applications of predictive control are based on linear models, which yield good results especially if they work around a duty point [18]. However, there are many applications where the region of operation and/or the degree of “non-linearity” of the system reduce the prediction capabilities of linear models, thus leading to poor controller performance. In such cases, Non Linear Model Predictive Control (NMPC) is a suitable option.

Although the number of applications of NMPC is limited [239–241], its potential is enormous. The possibility of dealing with nonlinear dynamics is the main advantage over MPC. However, developing precise nonlinear models from first principles may be a difficult task in many complex processes. Another disadvantage is that the optimiser solution in non-linear Predictive Control is a non-convex problem and a large computational effort may be required to obtain the solution. This is especially relevant when dealing with real time tasks.

Therefore, industrial control platforms with low computational power can not run non-linear predictive control strategies.

Due to the nonlinear nature of the fuzzy system, there are different solutions given to the FMPC optimisation problem. Branch and Bound [226, 242] or Genetic algorithms

[226, 243] are used by several authors, others linearise the TS fuzzy model in the operating point, solving a linear MPC problem [244–246]. A simpler scheme is used, designing multi-model in the TS fuzzy model [244, 246]. A study and comparison of fuzzy model based predictive control strategies will be covered in next sections. Two strategies of NMPC using neurofuzzy models will be presented, which, due to its low computational cost, will be suitable to implement in a typical industrial PLC of medium range, which is very common in industry.

5.1 Development and implementation of a FMPC for an industrial autoclave

The sterilisation of solid food in steam retorts has been chosen as a benchmark because this system exhibits highly nonlinear behaviour. In addition, the operation needs to be guided by the achievement of strict requirements on microorganisms thermal destruction while maintaining the product under acceptable organoleptic specifications. Such goals must be attained despite a number of undesirable disturbances acting within the process like sudden steam temperature shut down situations due to boiler overload (excessive steam demand from different retorts). The process is also subject to a considerable degree of parameter uncertainties and also, to some extent, a lack of accurate dynamic models (structural uncertainty), because many simplifications like spatial homogeneity or isotropy assumptions are considered in order to obtain tractable models. All these issues make the plant a good test bed for the illustration of the capabilities of the nonlinear predictive control strategies based on fuzzy models quoted before. Two different techniques have been developed using neurofuzzy models with GPC[22] and applied to control the thermal sterilisation process in steam retorts.

A schematic representation of the pilot plant is depicted in Figure 5.1. This unit belongs to IIM-CSIC (*Instituto de Investigaciones Marinas-Consejo Superior de Investigaciones Científicas*, Spain). The retort contains the product to be sterilised, usually consisting of a number of cans with the same specifications (geometry, size and type of food). Once in the retort, the load will be subjected to a time-temperature sterilisation profile previously “designed” to ensure a given lethality (a parameter related to the degree of reduction in a reference pathogen microorganism) while preserving the product quality as much as possible. The specified profile is enforced by regulating the flow of saturated steam produced by the boiler. This flow will first (in the venting stage, the first of the three stages of a sterilisation cycle) be used to remove the air present in the

retort, thus ensuring that the cans will be heated under the condition of saturated steam. This part of the operation, which is where sterilisation is really carried out, is known as *heating*. In real plants, during this stage, the product is usually kept at a given constant working temperature for a certain (predefined) time that will ensure the desired microbiological lethality. At the critical point (the coldest point/point of least lethality inside the product), lethality is defined as follows:

$$F_0 = \int_0^t 10^{\frac{T_{ref}^k - T(r_0)}{z}} dt \quad (5.1)$$

where z represents a kinetic parameter, T_{ref}^k its associated temperature and $T(r_0)$ represents the temperature at the critical point. It is worth noting that due to the exponential type relationship between temperature and lethality (equation 5.1), a fine temperature control is needed along this stage in order to avoid small disturbances that could result in a serious over-processing of the product.

Finally, and in order to avoid over-processing and therefore quality losses, the product needs to be cooled as fast as possible. Cooling water is employed during this part of the operation as a means of fast cooling while compressed air will be introduced in the retort in order to compensate for sharp pressure drops. A complete description of the sterilisation process and operation for different control schemes can be found in [167, 247–249]. From a control point of view, the system can be described as a MIMO plant, or more precisely, a set of MIMO plants representing the different stages of the process namely *venting*, *heating* and *cooling*. In the first two situations, inputs correspond with the positions of the steam, drain and purge valves, while the outputs are temperature and pressure inside the retort. The transition from venting to heating is detected by comparing current pressure measurements with the pressure corresponding to saturated steam which can be easily estimated from temperature measurements by applying the Antoine Law. In this way, when both variables (current pressure and saturated steam pressure) become equal, no air is present in the retort and heating may proceed. The system will be described by a set of ordinary differential equations, partial differential equations and algebraic equations, derived from mass and energy balances [167, 168].

5.1.1 Neurofuzzy model of the temperature inside the autoclave

In order to describe the temperature profile in the *heating* period, a neurofuzzy model was created. A real operation data set has been used in addition to simulation data

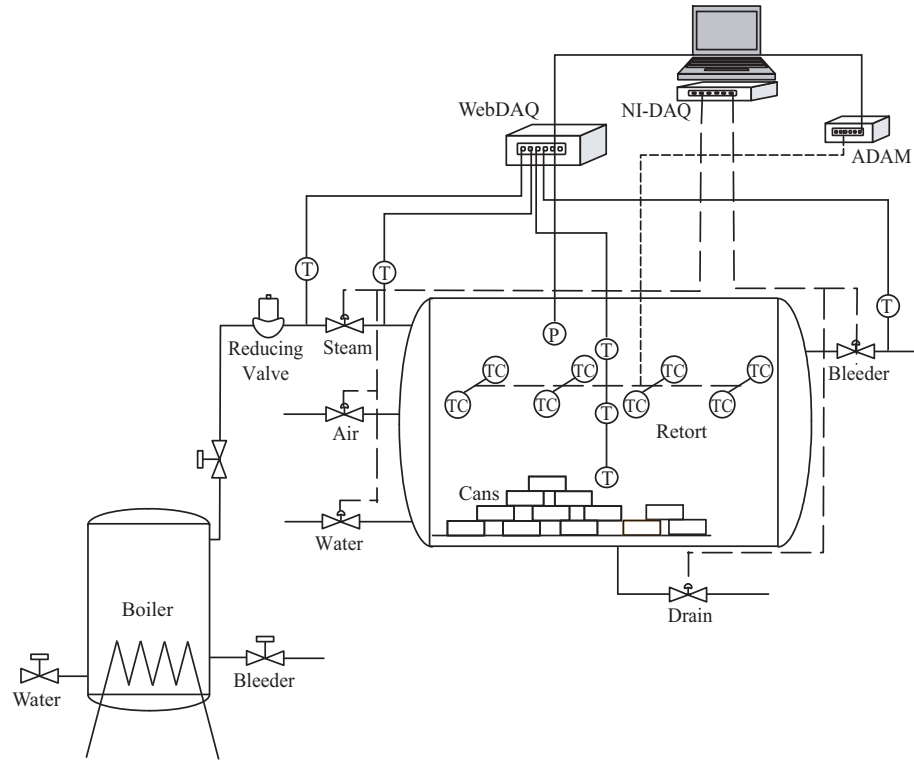


FIGURE 5.1: A prototype of thermal sterilisation in a steam retort [167]

obtained from a model derived from first principles and implemented in EcosimPro[®] [168, 250]. Inputs correspond with the positions of the steam valve (u_s), drain and purge valves (u_d, u_b), temperature of steam (T_s) and past samples (T_{-1}, T_{-2}) for the temperature inside the retort, while the output is the current temperature of autoclave.

The rules base obtained is:

- IF u_s IS in1mf1 AND T_s IS in2mf1 AND u_b IS in3mf1 AND u_d IS in4mf1 AND T_{-2} IS in5mf1 AND T_{-1} IS in6mf1

THEN:

$$T = 0.385u_s - 0.141T_s + 0.0584u_b + 0.0875u_d + 0.125T_{-2} + 0.732T_{-1} + 0.116$$

- IF u_s IS in1mf2 AND T_s IS in2mf2 AND u_b IS in3mf2 AND u_d IS in4mf1 AND T_{-2} IS in5mf2 AND T_{-1} IS in6mf2

THEN:

$$T = 0.0354u_s + 0.002T_s + 0.007u_b + 0.011u_d - 0.308T_{-2} + 1.272T_{-1} + 0.015$$

where in $imfj$ are the membership functions presented in figure 5.2. In figure 5.3 the

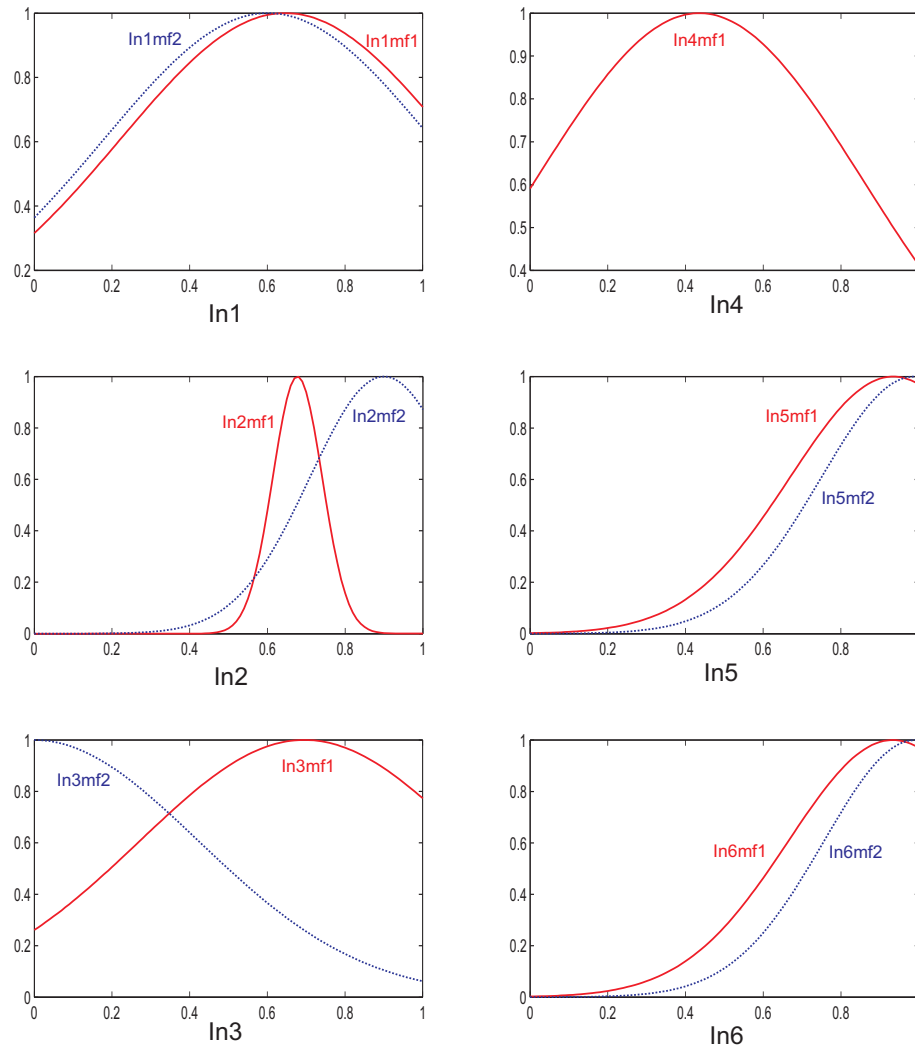


FIGURE 5.2: Membership functions of neurofuzzy system of autoclave

comparison between the model and the real data of the temperature in the autoclave is shown. The fuzzy model accurately captures the system behavior.

5.1.2 Fuzzy generalised Predictive Control (FGPC) of the temperature

As mentioned in the introduction, linear controllers may result in poor reference tracking when dealing with highly nonlinear systems. This fact will be illustrated in the following section by means of an experimental case study (the sterilisation of solid food in steam retorts). In these cases other alternatives, in which non-linear models in predictive control could be used, may result in suitable alternatives to the linear controllers. There are several proposals in this field [18, 251–253]. An interesting option is to use

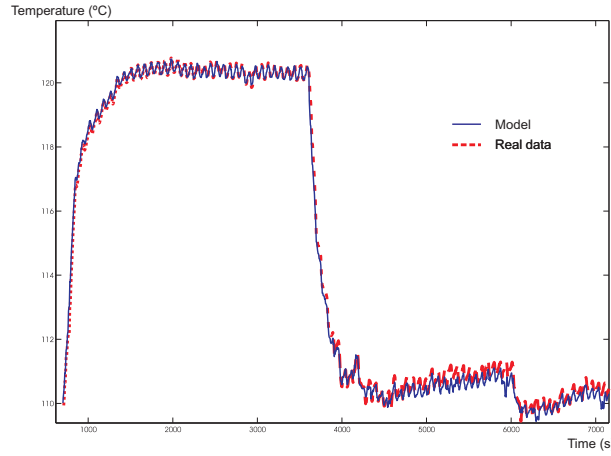


FIGURE 5.3: Validation of neurofuzzy model of the industrial autoclave

neurofuzzy models [225, 254–259]. Perhaps, the main issue in non-linear Predictive Control is how to obtain the optimiser solution, which consists in a non-convex problem and its resolution includes a high computational cost to solve in real time. During the last few years, several techniques have arisen to avoid the problems associated to determine the solution of the non-convex optimisation problem [18]. Moreover, closed loop stability must be guaranteed.

Two strategies based on the neurofuzzy model obtained for the autoclave are applied here. It is seen in chapter 2, the nonlinear model of the Takagi-Sugeno system, has a set of linear models equal to the number of rules. In this case, the linear models equations are:

$$T_1(t) = 0.385u(t-1) - 0.385u(t-2) + 0.125T(t-3) + 0.606T(t-2) + 1.732T(t-1) \quad (5.2)$$

$$T_2(t) = 0.035u(t-1) - 0.035u(t-2) - 0.308T(t-3) + 1.580T(t-2) + 2.273T(t-1) \quad (5.3)$$

where $T_i(t)$ is the temperature of autoclave for the local model i , and $u(t-j)$ is the position of steam valve at j previous samples.

A strategy proposed in [252, 257, 259, 260] involves calculating as many GPC controllers as linear models obtained in the neurofuzzy model, such that the controller output be (3.2):

$$u(k) = \sum_{j=1}^L w_j(k)u_j(k) \quad (5.4)$$

where L is the number of linear models and w_j is defined as

$$w_j(k) = \frac{\bar{\mu}_j(k)}{\sum_{j=1}^L \bar{\mu}_j(k)}, \quad \bar{\mu}_j(k) = \prod_{i=1}^n \mu_{ij}(k) \quad (5.5)$$

This procedure will be referred to as the FGPC1. The advantage of this technique is an easy and fast implementation and can be applied on a simple PLC. The main disadvantage of this strategy is that global optimum is not always found. However a local optimum is guaranteed for each rule or implication [260].

Another non-linear strategy based on neurofuzzy models (FGPC2) is proposed in [254]. A recurrent neurofuzzy model could be rewritten as a *Linear Time Variant* (LTV):

$$\bar{a}(z^{-1})y(k) = \bar{b}(z^{-1})u(k-d) + \xi(k) \quad (5.6)$$

Where d is a transport delay, $\xi(k)$ is a white noise sequence with null average and:

$$\begin{aligned} \bar{a}(z^{-1}) &= 1 - \bar{a}_1 z^{-1} - \bar{a}_2 z^{-2} - \dots - \bar{a}_{n_y} z^{-n_y} \\ \bar{b}(z^{-1}) &= \bar{b}_1 z^{-1} + \bar{b}_2 z^{-2} + \dots + \bar{b}_{n_u} z^{-n_u} \\ \bar{a}_i &= \sum_{j=1}^L w_j(k) a_{ij} z^{-i} \\ \bar{b}_i &= \sum_{j=1}^L w_j(k) b_{ij} z^{-i} \end{aligned}$$

The cost function defined in (1.9) can be expressed like:

$$J(k) = (Fy(k) + G\Delta u(k) + \Lambda - \Phi W)^T (Fy(k) + G\Delta u(k) + \Lambda - \Phi W) + (\lambda(z^{-1})\Delta u(k))^2 \quad (5.7)$$

Where:

$$\begin{aligned} F &= [f_d(z^{-1}) \quad f_{d+1}(z^{-1}) \quad \dots \quad f_{N_p}(z^{-1})]^T, \\ \Lambda &= \left[\sum_{\rho=1}^{d+N_u-1} g_{d,\rho} \Delta u(k-\rho) \quad \sum_{\rho=1}^{d+N_u} g_{d+1,\rho} \Delta u(k-\rho) \quad \dots \quad \sum_{\rho=1}^{N_p+N_u-1} g_{N_p,\rho} \Delta u(k-\rho) \right]^T \\ \Phi &= \text{diag}\{\delta_d \quad \delta_{d+1} \quad \dots \quad \delta_{N_p}\} \\ W &= [w(k+d) \quad w(k+d+1) \quad \dots \quad w(k+N_p)]^T \end{aligned}$$

The prediction horizon was chosen as $N_p = 3$ and, to reduce computational cost, the control horizon was reduced to $N_u = 1$. This leads to:

$$G^T (Fy(k) + \Lambda - \Phi W) + (G^T G + \lambda(z^{-1})\lambda_0)\Delta u^*(k) = 0 \quad (5.8)$$

To be able to simplify the control law, $\lambda_0^2 = \lambda > 0$ is chosen. Values $\lambda_1, \lambda_2, \dots, \lambda_{N_p}$ are adjusted such that they meet:

$$\Delta u^*(k) = \frac{G^T (\Phi W - Fy(k))}{G^T G + \lambda}. \quad (5.9)$$

A stability study of this control strategy applied to neurofuzzy model was made in [254].

5.1.3 Experimental results

The pilot plant described in the introduction will be now be used a case study to illustrate the performance of the different controllers derived in this work.

Using equal prediction and control horizons, based on a simulation tuning process, the values chosen for the prediction and control horizons were

$$N_p = N_u = 4$$

and the weighting parameter of the control action $\lambda = 0.6$, a GPC controller was obtained with the following law:

$$\begin{aligned} u(k) = & u(k-1) - 12.01y(k-1) + 21.68y(k-2) - 12.85y(k-3) + 2.30y(k-4) \\ & + 0.44w(k+1) + 0.27w(k+2) + 0.14w(k+3) + 0.04w(k+4) \end{aligned} \quad (5.10)$$

Where a soft approximation of the future reference trajectory has been used with $\alpha = 0.7$

$$w(t+k) = \alpha w(t+k-1) + (1-\alpha)r(t+k), k = 1, \dots, N_p \quad (5.11)$$

Figure 5.4 shows the closed loop response of the retort temperature (black line) with this controller. Four different steps have been introduced in the set point (green line). As illustrated in the figure 5.4, the controller is not only slow but it rarely reaches a good approximation to the set point.

As mentioned in the introduction, good controller performance is required in this type of process. In order to improve the closed loop response, the FGPC controllers have been

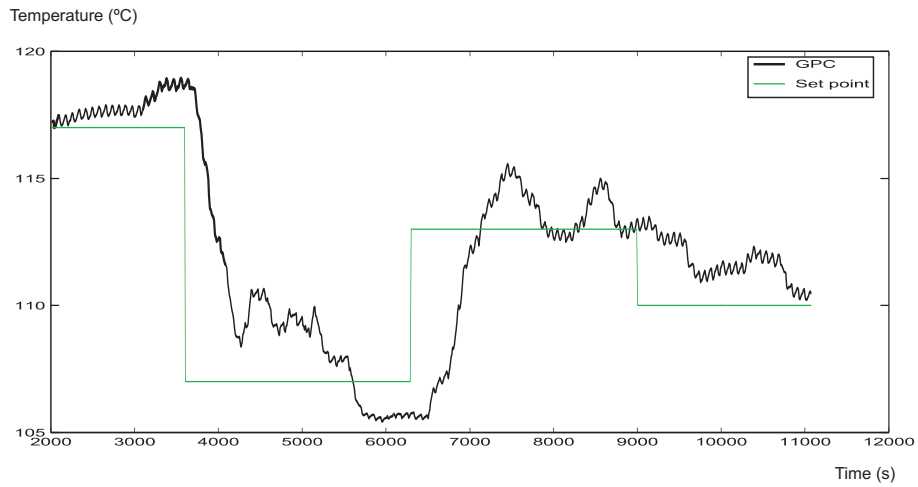


FIGURE 5.4: Performance of GPC

implemented. To carry out FGPC controllers proposed here, first of all it is necessary to change gaussian membership functions (see figure 5.2) to new triangular membership function defined by IEC61131-7. In figure 5.5, the new equivalent functions are shown. To have the output given in 5.4, FCL code is given in 5.1.4.

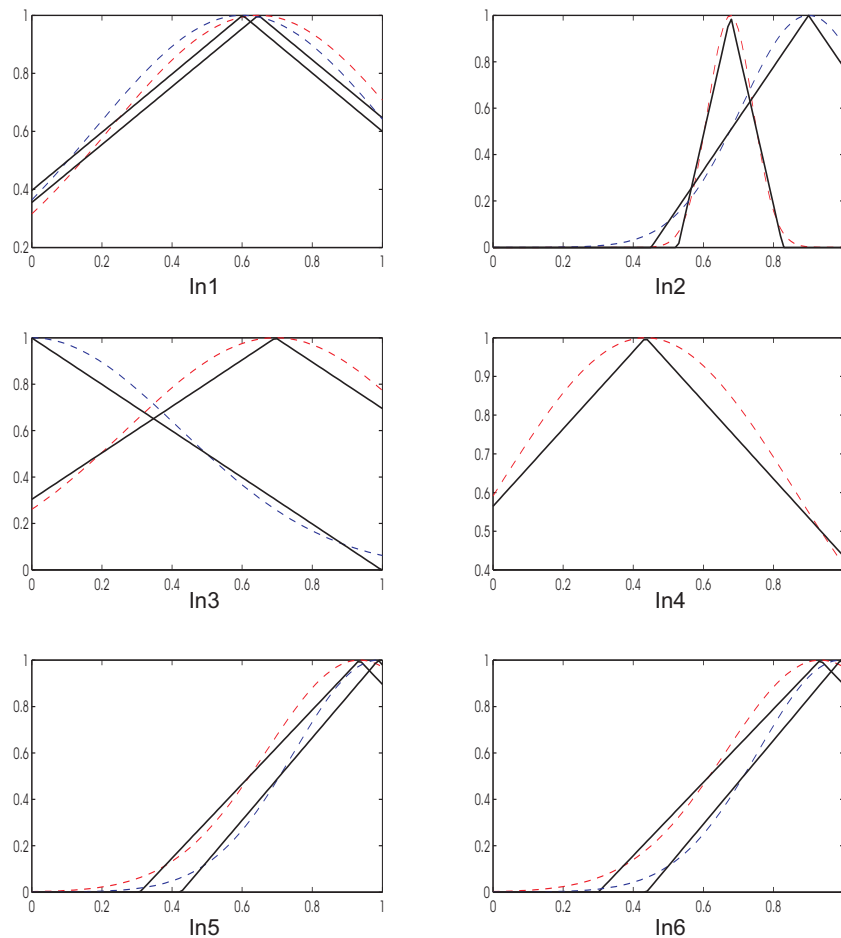


FIGURE 5.5: Equivalent triangular membership functions

The results for FGPC controller are represented in Figure 5.6. It is clear that this con-

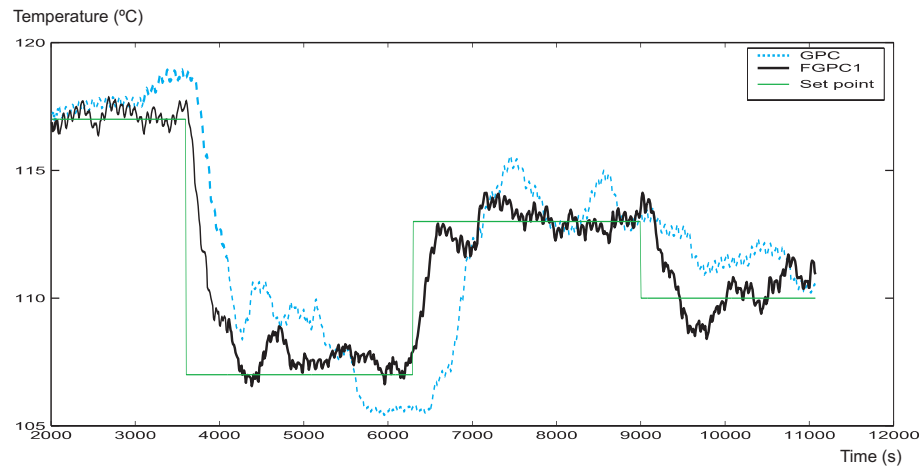


FIGURE 5.6: Comparison between Predictive Control strategies Non-linear vs linear

trol scheme is able to approach the retort temperature (black line) to the set point (green line) better than the GPC (turquoise line).

In a third test, the FGPC2 technique has also been applied to the pilot plant under the same conditions than the GPC and FGPC1. The black line in figure 5.7 corresponds

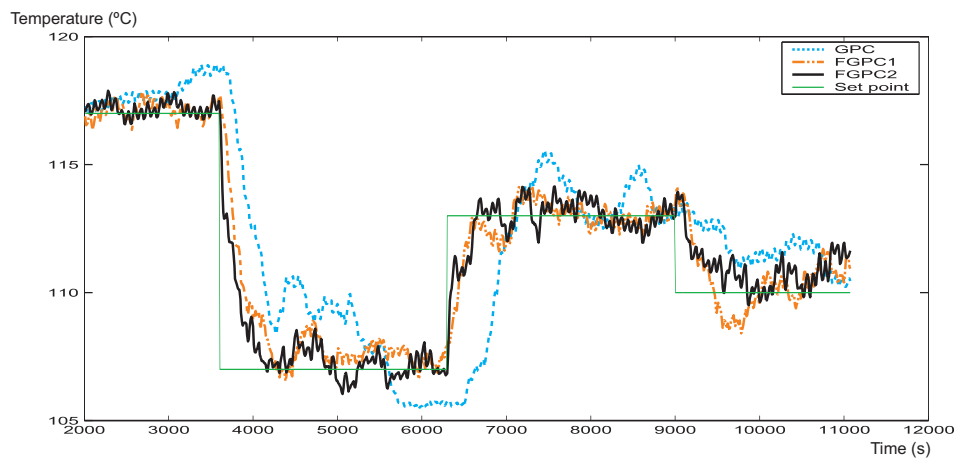


FIGURE 5.7: Comparison between strategies

with the closed loop (FGPC2) temperature response. On initial examination, it seems (although difficult to ascertain) that it improves the FGPC1 results (orange line). In order to obtain a more quantitative comparison between these two methods, the *Integral of Absolute Error* (IAE) (as a measurement of the tracking error) has been computed and presented in Table 5.1. The IAE for the FGPC2 is about a 20% lower than the IAE for the FGPC1 asserting that its performance is better.

A previous control strategy using a PI-type controller parameterized by means of the Internal Model Control (IMC) technique [261] has been presented in [167]. It is important to highlight that the FGPC2 presents a similar performance to the PI, but with

| GPC | FGPC1 | FGPC2 |
|---------------------|---------------------|---------------------|
| $1.8688 \cdot 10^4$ | $8.2837 \cdot 10^3$ | $6.8766 \cdot 10^3$ |

TABLE 5.1: IAE comparison

a saving of approximately a 70% in the energy consumption computed by using the index: $E_t = \int_t u_s(t) dt$.

5.1.4 FLC code accomplishing IEC 61131-7

As we saw in Chapter 2, the IEC61131-7 standard defines the language of fuzzy control for PLCs. The us_1 and us_2 variables, for FGPC1 strategy, must be previously calculated as the result of two GPC. Subsequently, the following code implements the fuzzy controller:

```

FUNCTION_BLOCK fmpc
  VAR_INPUT
    us    REAL; (* RANGE(0 .. 1) *)
    Ts    REAL; (* RANGE(0.7 .. 1) *)
    ub    REAL; (* RANGE(0 .. 1) *)
    ud    REAL; (* RANGE(0 .. 1) *)
    T_2   REAL; (* RANGE(0.4 .. 1) *)
    T_1   REAL; (* RANGE(0.4 .. 1) *)
  END_VAR
  VAR_OUTPUT
    T_0   REAL; (* RANGE(0.7460 .. 0.9928) *)
  END_VAR
  FUZZIFY us
    TERM us_m1 := (-0.3558, 0) (0.6463, 1) (1.6480, 0) ;
    TERM us_m2 := (-0.3949, 0) (0.6016, 1) (1.5980, 0) ;
  END_FUZZIFY
  FUZZIFY Ts
    Ts_m1 := (0.5273, 0) (0.6774, 1) (0.8275, 0) ;
    Ts_m2 := (0.4507, 0) (0.9000, 1) (1.3490, 0) ;
  END_FUZZIFY
  FUZZIFY ub
    TERM us_m1 := (-0.3038, 0) (0.6954, 1) (1.6950, 0) ;
    TERM us_m2 := (-0.9986, 0) (-0.0005, 1) (0.9975, 0) ;
  END_FUZZIFY

```

```

FUZZIFY ud
    TERM us_m1 := (-0.5639, 0) (0.4354, 1) (1.4350, 0) ;
    TERM us_m2 := (-0.5664, 0) (0.4314, 1) (1.4290, 0) ;
END_FUZZIFY
FUZZIFY T_2
    T2_m1 := (0.3081, 0) (0.9347, 1) (1.5610, 0) ;
    T2_m2 := (0.4255, 0) (0.9885, 1) (1.5520, 0) ;
END_FUZZIFY
FUZZIFY T_1
    T1_m1 := (0.3003, 0) (0.9335, 1) (1.5670, 0) ;
    T1_m2 := (0.4376, 0) (0.9917, 1) (1.5460, 0) ;
END_FUZZIFY
DEFUZZIFY TO
    TO_m1 := us1 ;
    TO_m2 := us2 ;
    METHOD: MoM;
END_DEFUZZIFY
RULEBLOCK first
    AND:MIN;
    ACCU:MAX;
    RULE 0: IF (us IS us_m1) AND (Ts IS Ts_m1) AND (ub IS ub_m1) AND
        (ud IS ud_m1) AND (T_2 IS T_2_m1) AND (T_1 IS T1_m1) THEN (T_0 IS TO_m1);
    RULE 1: IF (us IS us_m2) AND (Ts IS Ts_m2) AND (ub IS ub_m2) AND
        (ud IS ud_m2) AND (T_2 IS T_2_m2) AND (T_1 IS T1_m2) THEN (T_0 IS TO_m2);
END_RULEBLOCK
END_FUNCTION_BLOCK

```

5.2 FMPC with constraints. Implementation issues

One of the problems of NMPC when considering constraints is the heavy computational burden associated with its implementation. On the other hand, it is also difficult to analyze the stability of the closed-loop system [262]. In [263] a Takagi-Sugeno (TS) fuzzy model predictive controller, based on piecewise Lyapunov function (PLF), has been proposed to solve the above problems with FMPC. However this method may lead to a conservative performance of the controller. [264] presents an alternative way to reduce the conservatism using fuzzy Lyapunov function (FLF) which only needs to find an independent positive definite matrix for each submodel. More recently, [262, 265] have presented an extended-fuzzy Lyapunov function to improve the results. Notwithstanding the foregoing, the computing load of these methods prevents the implementation in

low-cost or industrial hardware as had been described in Chapter 1. There are some applications where the NN are adjusted to imitate the controller [266–268]. Once the NN is trained from the MPC controller, the amount of computation required is very small and can be compared to the online computation of the fast implementation methods. In this thesis, we propose a FIS for the same purpose. The use of Fuzzy system will permit the inclusion of adjusting parameters as another input variable. Let F_{λ_i} be a FIS obtained from a NMPC tuned by λ_i , composed by N_i rules R_j as was shown in 3.2:

R_j :

IF $x_1(k)$ is F_{1j}, \dots , and $x_n(k)$ is F_{nj} ,

THEN :

$$y_j(k) = a_j(z^{-1})y(k-1) + b_j(z^{-1})u(k-d) + \xi(k)$$

Figure 5.8 shows the procedure to adjust F_{λ_i} from the NMPC. Changing parameter λ_i to reach several control performance modes, e.g. *aggressive*, *moderate*, *slow*, we will obtain as many fuzzy systems as λ_i chosen. The union of fuzzy systems $\{F_{\lambda_1}, F_{\lambda_2}, \dots, F_{\lambda_p}\}$ will result another fuzzy system with a new input variable λ_i with $N = N_1 + N_2 + \dots + N_p$ rules:

R_j :

IF $x_1(k)$ is F_{1j}, \dots , and $x_n(k)$ is F_{nj} , and λ_j is Λ_{rj}

THEN :

$$y_j(k) = a_j(z^{-1})y(k-1) + b_j(z^{-1})u(k-d) + \xi(k) \quad (5.12)$$

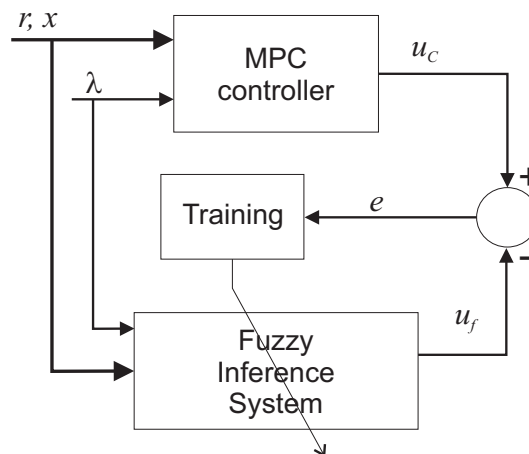


FIGURE 5.8: Scheme to obtain a fast FIS from another controller

5.3 Conclusion of the chapter

In this chapter FNMPC has been presented as an alternative to NMPC, saving computational burden. Two different schemes have been implemented in a real industrial plant, using low computational cost hardware with FCL. The results have been compared with a linear GPC, obtaining better performance with no significant programming effort and computational resources.

NMPC subject to constraint has also been introduced, and a new technique to carry out a fast implementation providing an adjusting parameter has been proposed. Both the complexity of the system and the requirement of a higher precision in the parameter setting can cause an explosion of rules in the fuzzy model. Reasonably managing a FIS for use in control, involves applying a rules reduction technique. In the following chapters we will introduce a novel technique for complexity reduction in fuzzy systems based on their structure.

Chapter 6

Complexity reduction in fuzzy systems using Functional Principal Component Analysis

In this chapter a novel technique to reduce complexity in fuzzy models will be described. The aim of such reduction is the suitability to control systems which run on low capability hardware platforms. Firstly, in section 6.1, the state of the art in complexity reduction of fuzzy systems will be explored. Secondly in sections 6.2 and 6.3 the *Functional Principal Component Analysis* (FPCA) will be described. Section 6.4 will show the main contribution of this chapter: the application of FPCA to reduce complexity of a fuzzy inference system structures. Section 6.5 will describe example applications before drawing conclusions on this chapter.

6.1 Complexity reduction in fuzzy systems

The ability to build fuzzy logic applications for control problems has been hindered by the well-known problem of combinatorial rules explosion, causing complexity in modeling. The existence of redundant rules may also cause performance degradation of the FIS [123].

There has been an increased interest in the issues of complexity of fuzzy systems over recent years. There are a remarkable number of methods aimed at reducing the complexity of fuzzy systems. Most of them are based on systematic and heuristic methods

[269–272], others with analytic approach, are practically unapplicable when the number of inputs is large, [269, 273–278]. In [279] there is a classification methods for Mamdani and TS systems. A simplification method for direct fuzzy controllers is presented in [280, 281].

In this chapter, a new technique to reduce the number of rules will be presented. It is based on FPCA, one of the methods of functional analysis. This method provides a systematic approach to the rule reduction. After its application, the new FIS will have a lower number of rules. We will demonstrate that the implementation of this technique is at the detriment of interpretability of the system's background. The new fuzzy system will present a non-conventional antecedent set which won't however be an issue for deployment in control systems.

6.2 Principal Component Analysis

Principal Components Analysis (PCA) is well known in the field of multivariate statistical analysis. Using PCA, we reduce the dimensionality of the space variables. The idea behind PCA is to find the subspace where the data have a high covariance. Supposing r variables and N real samples, a real data set is represented by:

$$\begin{aligned}
 & x_{11}\vec{e}_1 + x_{12}\vec{e}_2 + \dots + x_{1r}\vec{e}_r \\
 & x_{21}\vec{e}_1 + x_{22}\vec{e}_2 + \dots + x_{2r}\vec{e}_r \\
 & \quad \quad \quad \vdots \quad \quad \dots \\
 & x_{N1}\vec{e}_1 + x_{N2}\vec{e}_2 + \dots + x_{Nr}\vec{e}_r
 \end{aligned} \tag{6.1}$$

Presenting the data in matrix format:

$$\mathbf{x} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1r} \\ x_{21} & x_{22} & \dots & x_{2r} \\ \vdots & \vdots & \dots & \vdots \\ x_{N1} & x_{N2} & \dots & x_{Nr} \end{pmatrix} \tag{6.2}$$

\vec{e}_i are the vectors generators of subspace V . The aim is to find a new basis vector (y_1, y_2, \dots, y_m) to define a new subspace containing the maximum information of the actual data:

$$\begin{aligned}
y_1 &= w_{11}x_1 + w_{12}x_2 \dots + w_{1r}x_r \\
y_2 &= w_{21}x_1 + w_{22}x_2 \dots + w_{2r}x_r \\
&\vdots \\
y_m &= w_{m1}x_1 + w_{m2}x_2 \dots + w_{mr}x_r
\end{aligned} \tag{6.3}$$

Where $m < r$. If $\mu_y = E(\mathbf{y})$ is the expected value of \mathbf{y} . It demonstrates that:

$$\mu_y = E(\mathbf{w}^T \mathbf{x}) = \mathbf{w}^T E(\mathbf{x}) \tag{6.4}$$

and the covariance matrix of \mathbf{y} is equal to:

$$\mathbf{C}_y = E\{(\mathbf{y} - \mu_y)(\mathbf{y} - \mu_y)^T\} = \mathbf{w}^T \mathbf{C}_x \mathbf{w} \tag{6.5}$$

To obtain the subspace with maximum variability in the data, we calculate the covariance matrix of \mathbf{y} (\mathbf{C}_y), imposing the orthonormality constraint on it:

$$\mathbf{w}^T \mathbf{w} = I \tag{6.6}$$

Which we have to optimise:

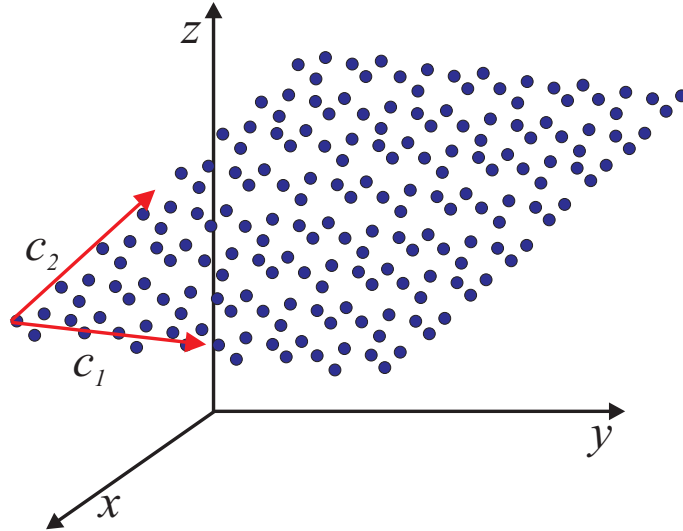
$$\mathbf{w}^T \mathbf{C}_x \mathbf{w} - \lambda(\mathbf{w}^T \mathbf{w} - I) \tag{6.7}$$

differentiating and equating to zero:

$$(\mathbf{C}_x - \lambda I)\mathbf{w} = 0 \tag{6.8}$$

The problem is reduced to calculating the eigenvectors of \mathbf{C}_x . Those associated with the most significant eigenvalues, which components will form the subspace where the data have highest variability.

In order to choose the number of principal components, a criterion for choosing the eigenvalues may be the use of an index of variability, defined as follows:

FIGURE 6.1: Principal components: C_1 and C_2 .

$$\frac{\sum_{i=1}^l \lambda_i}{\sum_{i=1}^n \lambda_i} \geq \nu \quad (6.9)$$

Where ν is the degree of desired information, or variability index (1 value, means that we use all the eigenvalues).

6.3 Functional Principal Component Analysis

The PCA works in a vector space. If we work in a space of functions, the analysis will be the FPCA. Let $f_1(x), f_1(x), \dots, f_n(x)$ be functions in separable Hilbert space endowed with inner product:

$$\langle f_i | f_j \rangle = \int_0^X f_i(x) f_j(x) dx \quad \forall f_i, j \in L^2[0, X] \quad (6.10)$$

If each function $f_i(x)$ may be decomposed in:

$$f_i(x) = \sum_{l=1}^L c_{il} \theta_l(x) = \mathbf{c}_i^T \Theta(x) \quad (6.11)$$

The mean and covariance functions of f_i , will be:

$$\bar{f}(x) = E(f(x)) = \bar{\mathbf{c}}^T \Theta(x) \quad (6.12)$$

$$\text{Cov}[f(x), f(s)] = \Theta(x)^T \text{cov}(\mathbf{C}) \Theta(s) \quad (6.13)$$

Where $\mathbf{C} = \{c_{il}, i = 1, \dots, n, l = 1, \dots, L\}$.

We define the covariance operator as:

$$C(f(x)) = \int_0^X \text{Cov}[f(x), f(s)] f(s) ds, \quad \forall f \in L^2[0, X], \forall x, s \in [0, X] \quad (6.14)$$

Where the kernel $\text{Cov}[f(x), f(s)]$ is the covariance function.

The covariance operator is positive, selfadjoint and compact [282], thus, using Mercer's Theorem, we may write:

$$\text{Cov}[f(x), f(s)] = \sum_{i=1}^{\infty} \lambda_i \xi_i(x) \xi_i(s), \quad \forall x, s \in [0, X] \quad (6.15)$$

where $\lambda_1 > \lambda_2 > \dots > 0$ is an enumeration of the eigenvalues of C , and the corresponding orthonormal eigenfunctions are ξ_1, ξ_2, \dots . Thus, they form a complete orthonormal set of solutions of the Fredholm equation:

$$\int_0^X \text{Cov}[f(x), f(s)] \xi_i(s) ds = \lambda_i \xi_i(x) \quad (6.16)$$

We can formulate the expression 3.2 as:

$$y(\mathbf{x}) = \tilde{g}_0(\mathbf{x}) + \tilde{g}_1(\mathbf{x})x_1 + \dots + \tilde{g}_n(\mathbf{x})x_n \quad (6.17)$$

Where:

$$\tilde{g}_i(\mathbf{x}) = \sum_{j=1}^N a_j(\mathbf{x}) \cdot g_{ji} \quad (6.18)$$

And the vector of functions $\tilde{\mathbf{g}}$ is:

$$\tilde{\mathbf{g}}(\mathbf{x}) = \begin{bmatrix} \tilde{g}_0(\mathbf{x}) \\ \tilde{g}_1(\mathbf{x}) \\ \vdots \\ \tilde{g}_n(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} g_{10} & g_{20} & \dots & g_{N0} \\ g_{11} & g_{21} & \dots & g_{N1} \\ & \vdots & & \\ g_{1n} & g_{2n} & \dots & g_{Nn} \end{bmatrix} \cdot \begin{bmatrix} a_0(\mathbf{x}) \\ a_1(\mathbf{x}) \\ \vdots \\ a_N(\mathbf{x}) \end{bmatrix} \quad (6.19)$$

$$\tilde{\mathbf{g}}(x) = \mathbf{G} \cdot \mathbf{a}(x)$$

The mean and covariance functions of $\tilde{\mathbf{g}}(x)$, are:

$$\begin{aligned} E[\tilde{\mathbf{g}}(x)] &= E[\mathbf{g}^T] \cdot \mathbf{a}(x) = \bar{\mathbf{g}}^T \cdot \mathbf{a}(x) \\ Cov[\tilde{\mathbf{g}}(x), \tilde{\mathbf{g}}(s)] &= \mathbf{a}(x)^T cov(\mathbf{G})\mathbf{a}(s) \end{aligned} \quad (6.20)$$

We have to solve the equation (6.16), to obtain the FPCA of these functions. We suppose that the eigenfunctions are

$$\xi(x) = \mathbf{a}(x)^T \cdot \mathbf{b} \quad (6.21)$$

Thus, taking in account (6.20):

$$\begin{aligned} \int_0^X Cov[\tilde{\mathbf{g}}(x), \tilde{\mathbf{g}}(s)] \cdot \xi(s) ds &= \int_0^X \mathbf{a}(x)^T cov(\mathbf{G})\mathbf{a}(s) \cdot \mathbf{a}(s)^T \cdot \mathbf{b} ds \\ &= \mathbf{a}(x)^T cov(\mathbf{G}) \cdot \mathbf{W} \cdot \mathbf{b} \\ cov(\mathbf{G}) \cdot \mathbf{W} \cdot \mathbf{b} &= \lambda \cdot \mathbf{b} \end{aligned} \quad (6.22)$$

Where:

$$\mathbf{W} = \int_0^X \mathbf{a}(s) \cdot \mathbf{a}(s)^T ds \quad (6.23)$$

The functions $\xi(x)$ are orthogonal, then $\langle \xi_i(x), \xi_j(x) \rangle = b_i^T \cdot \mathbf{W} \cdot b_j = 0$. Matrix \mathbf{W} is symmetric by definition, thus, defining $\mathbf{u} = \mathbf{W}^{\frac{1}{2}} \mathbf{b}$,

$$\mathbf{W}^{\frac{1}{2}} \cdot cov(\mathbf{G}) \cdot \mathbf{W}^{\frac{1}{2}} \cdot \mathbf{u} = \lambda \cdot \mathbf{u} \quad (6.24)$$

We are left with solving a symmetric eigenvalue problem. Afterward, using a variability criteria, we can choose a new subspace using a new base of eigenfunction whose eigenvalues have enough significance, for instance 6.9, where $v \in [0, 1]$ is the variability index ($v = 1$ corresponding to the maximum variability obtained in the new space, i.e. the new subspace has the same dimension of the original space). N is the dimension of the original space and R is for the new reduced subspace.

6.4 FPCA for fuzzy systems

Having applied FPCA on 6.3, we obtained a new subspace of functions

$$\gamma(x) = \left[\gamma_1(x) \quad \gamma_2(x) \quad \dots \quad \gamma_n(x) \right]^T \quad (6.25)$$

such that:

$$\tilde{\mathbf{g}}(x) = \begin{bmatrix} \tilde{g}_0(x) \\ \tilde{g}_1(x) \\ \vdots \\ \tilde{g}_n(x) \end{bmatrix} = \begin{bmatrix} h_{10} & h_{20} & \dots & h_{R0} \\ h_{11} & h_{21} & \dots & h_{R1} \\ & \vdots & & \\ h_{1n} & h_{2n} & \dots & h_{Rn} \end{bmatrix} \cdot \begin{bmatrix} \xi_0(x) \\ \xi_1(x) \\ \vdots \\ \xi_R(x) \end{bmatrix}$$

$$\tilde{\mathbf{g}}(x) = \mathbf{H} \cdot \xi(x) \quad (6.26)$$

Comparing with (6.19), the new R rules ($R < N$) of the fuzzy system will be,

Rule R_j :

IF x_1 is Γ_{x_1j} , ..., and $x_n(k)$ is Γ_{x_nj} ,

THEN: $y_j = h_{0j} + h_{1j}x_1 + \dots + h_{nj}x_n$

Where Γ_{x_ij} is the fuzzy set respective to $x_i(k)$ on the rule j . And $\mu_{\Gamma_{ij}}(x_i)$ will be the new membership degree of x_i to the set Γ_{ij} . For each rule, we must have:

$$\sigma_j(x) = \mu_{\Gamma_{1j}}(x_1) \cdot \mu_{\Gamma_{2j}}(x_2) \cdot \dots \cdot \mu_{\Gamma_{nj}}(x_n) \quad (6.27)$$

Or

$$\sigma_j(x) = \min\{\mu_{\Gamma_{1j}}(x_1), \mu_{\Gamma_{2j}}(x_2), \dots, \mu_{\Gamma_{nj}}(x_n)\} \quad (6.28)$$

And

$$\xi_i(x) = \frac{\sigma_i(x)}{\sum_{i=1}^R \sigma_i(x)} \quad (6.29)$$

The new Γ_{ij} should be *prototypical* in order to be implemented in a controller, using the standard Fuzzy Control Language (FCL), i.e. with membership functions showed in fig. 2.2. This implies that the $\xi_i(x)$ should be $0 < \xi_i(x) < 1$ and convex. The IEC61131-7 standard permits non-prototypical shapes for the membership function [11] defined by a set of points. PLCs which use the norm (e.g. [101]) don't, to this day, include this functionality but they have functions to operate with the fuzzification result, permitting calculation of the eigenfunctions (eq. 6.21).

6.5 Illustrative examples

6.5.1 Pilot plant

To illustrate the ideas developed in this chapter we will examine two examples. The first one is a pilot plant site in the Department of System Engineering and Automatic Control of the University of Seville 6.2. The plant is used to emulate exothermic chemical reactions based on temperature changes. It has previously been used as a benchmark for control by researchers [283]. The main elements of the pilot plant are the reactor, the heat exchanger, the cooling jacket and the valve to manipulate the flow rate through the cooling jacket 6.3.



FIGURE 6.2: Pilot plant of the Department of System Engineering and Automatic Control of University of Seville

The emulated chemical reaction represents a refinement process. At the same flow at the inlet of the reactor outlet and a constant volume, the model of the chemical reaction

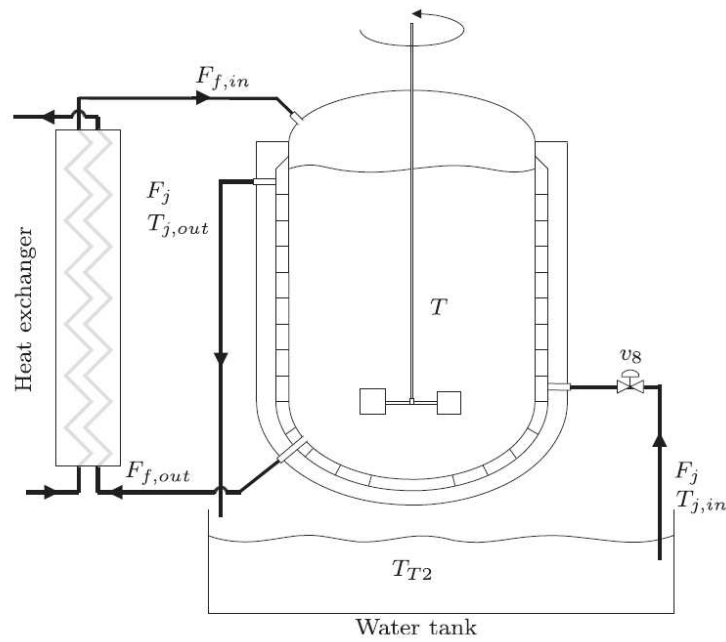


FIGURE 6.3: Diagram of the pilot plant

TABLE 6.1: Parameters and variables of the mathematical model of the pilot plant

| Parameter | Value | Unit |
|---|------------------------|-----------------|
| C_p (Specific heat capacity) | 4.18 | $KJ/K \cdot Kg$ |
| ΔH (Molar reaction heat) | -105.57 | KJ/mol |
| V (Volume of the reactor content) | 25 | l |
| M (Mass of the reactor content) | 25 | Kg |
| $C_{A,in}$ (Reactant concentration in the feed) | 1.2 | mol/l |
| F_j (Cooling jacket flow rate) | 0.05 | l/s |
| k_0 (Constant) | 1.265×10^{17} | l/mol |
| E/R (Constant) | 13550 | K |

can be defined as:

$$\frac{dT}{dt} = -\frac{F_j}{V}(T_{j,in} - T_{j,out}) + \frac{(-\Delta H)V}{M \cdot C_p} k_0 e^{-E/(RT)} C_A^2 \quad (6.30)$$

$$\frac{dC_A}{dt} = \frac{F_f}{V}(C_{A,in} - C_A) - k_0 e^{-E/(RT)} C_A^2 \quad (6.31)$$

Where T is the temperature into the reactor, $T_{j,in}$ the inlet temperature of the cooling jacket fluid, C_A denotes the reactant concentration in the reactor. The other variables and parameters can be seen in 6.1 Using real data, the system can be modeled by a Neurofuzzy System. Being a discrete model, in order to capture the dynamics of the system, two input variables have been chosen: The valve previous position $V(k-1)$

and the previous sampled temperature $T(k-1)$. The output is the actual temperature $T(k)$. Figure 6.4 shows the membership functions. Using a training learning method, we obtain the following rules:

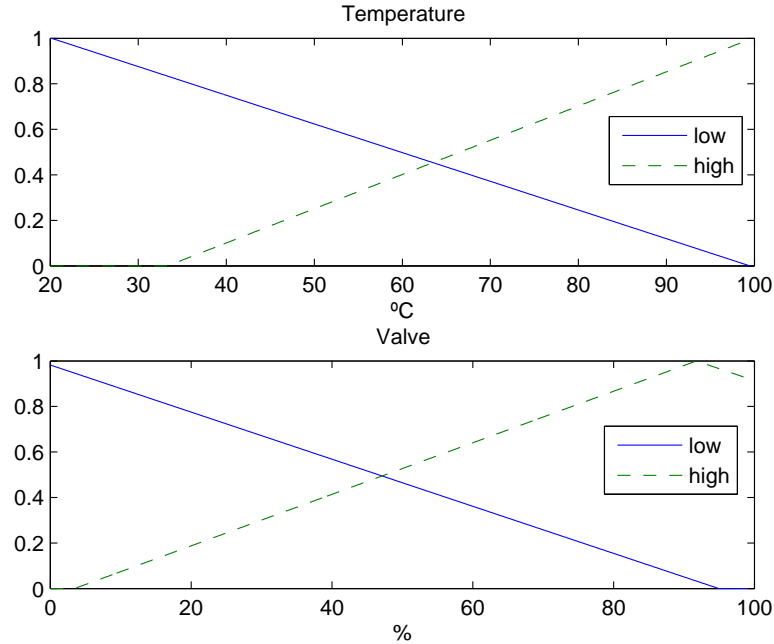


FIGURE 6.4: Membership functions for the Pilot plant Fuzzy model

- if $T(k-1)$ is LOW and $V(k-1)$ is LOW
then $T(k) = 1.0045T(k-1) + 0.0005V(k-1) - 0.0898$
- if $T(k-1)$ is LOW and $V(k-1)$ is HIGH
then $T(k) = 1.0006T(k-1) + 0.0005V(k-1) - 0.0571$
- if $T(k-1)$ is HIGH and $V(k-1)$ is LOW
then $T(k) = 1.0037T(k-1) - 0.0005V(k-1) - 0.3197$
- if $T(k-1)$ is HIGH and $V(k-1)$ is HIGH
then $T(k) = 1.0002T(k-1) - 0.0020V(k-1) - 0.0158$

To perform the FPCA on the model, we proceed as follows:

FPCA algorithm: (6.5.1)

1. Calculation of W from eq. 7.22

2. $W^{1/2}$ obtained through Cholewsky decomposition
3. Solving the eigenvalue problem 6.24
4. Getting $b = W^{-1/2}u$

The new system is:

$$\tilde{\mathbf{g}}(x) = \begin{pmatrix} -0.04154 \\ 0.00001 \\ 0.00531 \end{pmatrix} \cdot \xi(x) \quad (6.32)$$

and

$$\xi(x) = \mathbf{a}(x)^T \cdot \begin{pmatrix} -0.01020 \\ -0.00998 \\ -0.01149 \\ -0.00976 \end{pmatrix} \quad (6.33)$$

A comparison between systems is shown in figure 6.5.

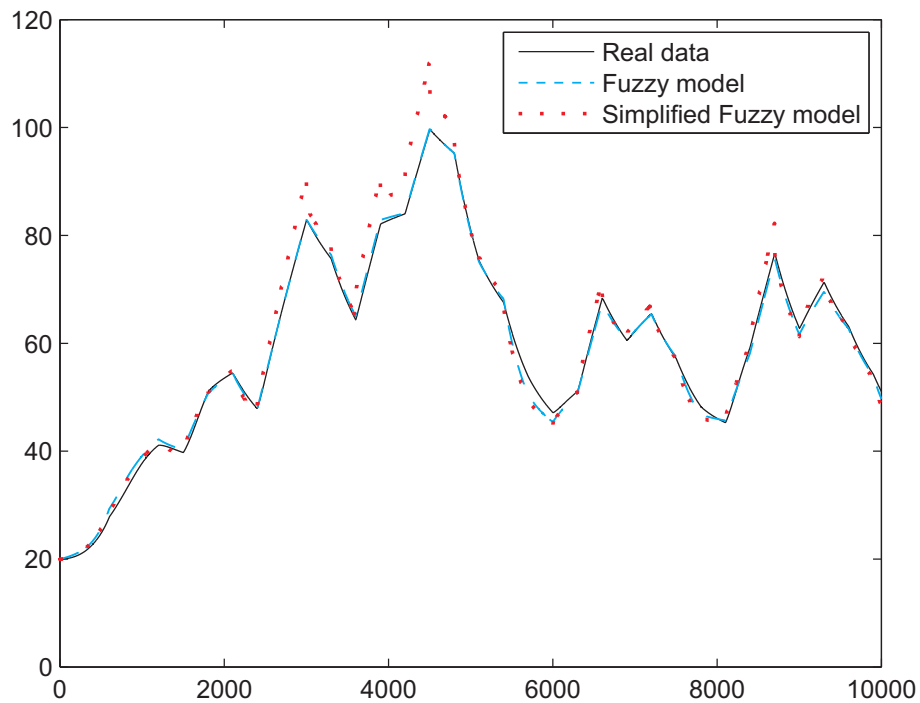


FIGURE 6.5: Pilot plant: Comparison between original, Fuzzy and simplified Fuzzy system

6.5.2 Mechanical system

Another interesting example is the mechanical system shown in figure 6.6. It could be a simple manipulator with only one joint. The system is moved by an electrical motor which provides a torque T_u in order to move a bar an angle θ . If we consider all the mass (m) concentrated at the end of the bar, the equation that describes the system is:

$$m\ddot{\theta}l^2 + B\dot{\theta} + mg \sin \theta = T_u \quad (6.34)$$

For simulation the parameters will be: $g = 9.8m/s^2$, $l = 1m$, $B = 1Kgm^2/s$, $m = 1Kg$. As we can observe in figure 6.6 the system has a non-linearity due to $\sin \theta$. Linearizing

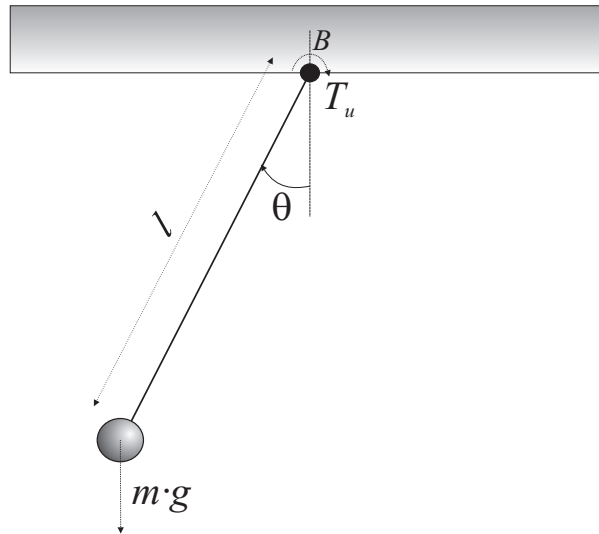


FIGURE 6.6: Mechanical system

around an equilibrium point, we could model the system as

$$\ddot{\theta} = -a\dot{\theta} - b\theta + T_u \quad (6.35)$$

Where a, b are parameters depending on the operating point (θ_0). It is a second order linear system. In order to build a Fuzzy system, we use four variables in discrete mode, to get the dynamics of a 2^{nd} order system: $T_u(k-2), T_u(k-1), \theta(k-2), \theta(k-1)$. Providing data sets for training and checking, the FIS obtained is defined by the membership function depicted in figure 6.7. Taking small steps to the input (torque), we can model the response as second order system 6.35, different for each operating point determined

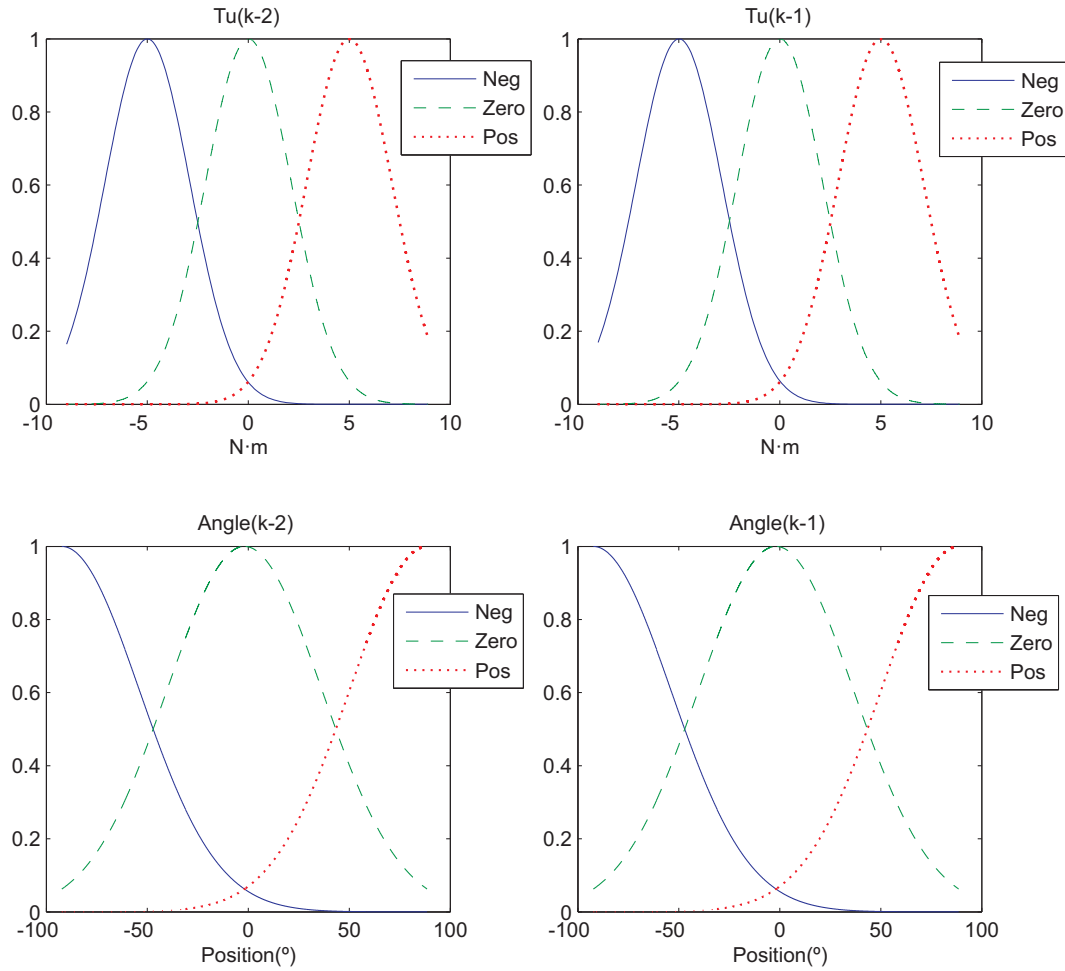


FIGURE 6.7: Membership functions for the mechanical system FIS

by the position of the mechanism. Doing this in nine areas, we have nine linear systems:

$$\begin{aligned} \theta_1(k) &= 0.0037T(k-1) + 0.0467T(k-2) - 0.9705\theta(k-1) + 1.9705\theta(k-2) \\ \theta_2(k) &= -0.0016T(k-1) + 0.0525T(k-2) - 0.9704\theta(k-1) + 1.9645\theta(k-2) \\ \theta_3(k) &= -0.0001T(k-1) + 0.0508T(k-2) - 0.9704\theta(k-1) + 1.9628\theta(k-2) \\ \theta_4(k) &= -0.0003T(k-1) + 0.0511T(k-2) - 0.9704\theta(k-1) + 1.9621\theta(k-2) \\ \theta_5(k) &= -0.0003T(k-1) + 0.0508T(k-2) - 0.9705\theta(k-1) + 1.9619\theta(k-2) \\ \theta_6(k) &= -0.0003T(k-1) + 0.0510T(k-2) - 0.9704\theta(k-1) + 1.9621\theta(k-2) \\ \theta_7(k) &= -0.0002T(k-1) + 0.0509T(k-2) - 0.9704\theta(k-1) + 1.9629\theta(k-2) \\ \theta_8(k) &= -0.0008T(k-1) + 0.0515T(k-2) - 0.9704\theta(k-1) + 1.9650\theta(k-2) \\ \theta_9(k) &= 0.00065T(k-1) + 0.0501T(k-2) - 0.9704\theta(k-1) + 1.9705\theta(k-2) \end{aligned}$$

where $\theta_i(k)$ is the angle variation for the local model i , and $T(k-j)$ is the variation of the applied torque. establishing rules with variable angle as antecedent, we can model

the mechanical system with minimum error, as is shown in fig 6.8, obtaining a RMSE of $\pm 0.1064^\circ$ over 3334 samples. In order to simplify the system applying a FPCA, the

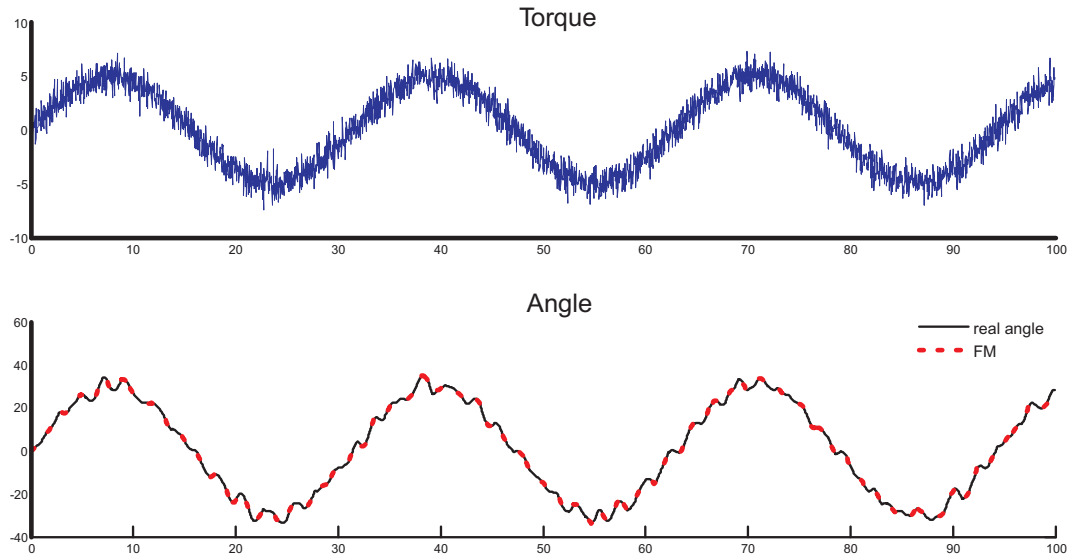


FIGURE 6.8: Validation of the fuzzy model for the mechanical system

procedure seen in 6.5.1 will be used. It is observed that the first eigenvalue contains almost all the variability of the system. Thus, the new simplified system will have just one rule and its structure is given by:

$$\tilde{\mathbf{g}}(x) = \begin{pmatrix} 0 \\ 0.0198 \\ -0.3788 \\ 0.7669 \\ -0.0034 \end{pmatrix} \cdot \xi(x) \quad (6.36)$$

And

$$\xi(x) = \mathbf{a}(x)^T \cdot \begin{pmatrix} 0.0425 \\ 0.0433 \\ 0.0433 \\ 0.0433 \\ 0.0433 \\ 0.0433 \\ 0.0434 \\ 0.0435 \\ 0.0444 \end{pmatrix} \quad (6.37)$$

In the figure 6.9 we can distinguish differences between the fuzzy and simplified fuzzy models. However, with an RMSE of $\pm 2.1692^\circ$ over 3334 samples, it is reasonable to use the simplified model for control or simulation.

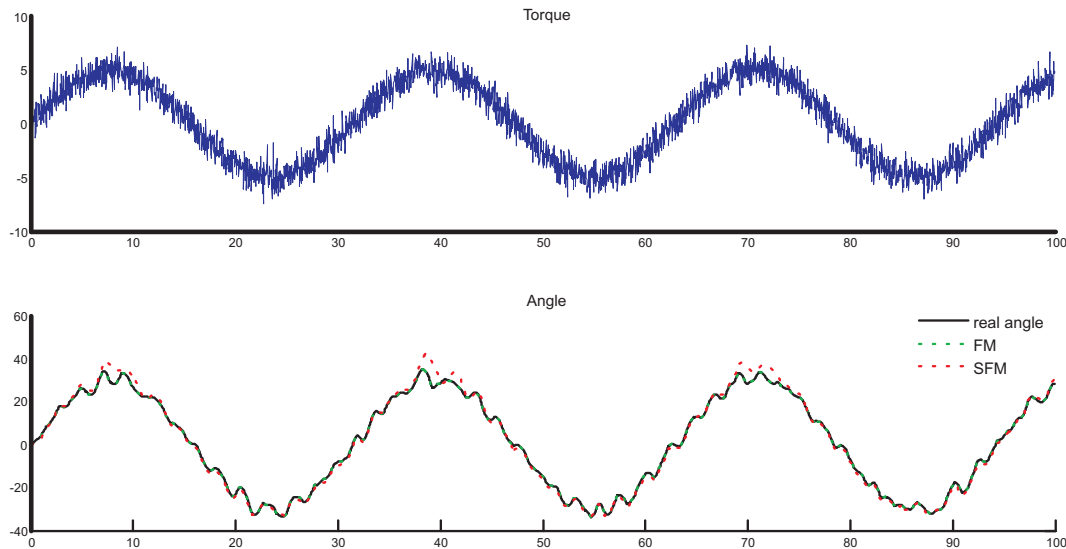


FIGURE 6.9: Mechanical system: Comparison between original, Fuzzy and simplified Fuzzy system

6.6 Conclusion of the chapter

In this chapter a new analytic technique has been presented in order to reduce the complexity of TS fuzzy systems. The main contribution of the thesis is the application of FPCA to the Hilbert space of functions defined by the rule base. Defining a covariance operator, the technique permits to reduce the space to a subspace where the operator's eigenvalues are bigger. There have been two examples where the technique has been applied successfully.

The problem with this technique is the lack of interpretability in the fuzzy system, due to the algebraic combination before aggregation. If we focus on the use in predictive control, this is not a problem. Even predictive controllers with explicit solutions that could be implemented in industrial devices according to IEC61131-3 [7] and IEC61131-7 [11], there may be the possibility of non-prototypical membership functions or algebraic operations after fuzzification.

Chapter 7

FPCA to simplify MPC implementation

Multivariate Statistics is used in control engineering for many years [284]. Singular Value Decomposition (SVD) techniques such as PCA has been used in control engineering for sensor fault detection [285], variable decoupling [286] and modelling [287, 288]. Dimensionality reduction [289] is the main feature that takes advantage of these techniques.

In this chapter, PCA and FPCA techniques studied in Chapter 6 will be used to simplify the control implementation. The first section of this chapter will show a PCA application for input space reduction, simplifying the control scheme. In section 7.2 the technique of the chapter 6 will be applied to simplify a FMPC and make it implementable in nonlinear systems with low capability hardware. The same technique will be applied in section 7.3 and 7.4, in order to reduce complexity of PWA systems and implement FMPC with constraints.

7.1 Dimensionality reduction of input variables space

In a *Multiple Inputs Single Output* (MISO) system with n inputs, the values that acquire the inputs can be considered as vectors in a space of dimension n . The control problem is to determine the sequence of vectors that produce the desired output. The application of PCA may reduce the inputs space dimension, simplifying the control problem. If the new input space has just 1 dimension, the system become *Single Input Single Output*

(SISO), reducing the complexity of the control problem, using only one manipulated variable

In the plant showed in section 3.3.2, after applying a PCA over the input variables, the first principal component accounts by itself for almost 80% of the information. The idea is to use this component as a new *virtual* input of the system keeping the pressure as the controlled output (see Figure 7.1). The control action obtained using this technique is then projected into the original axis in order to obtain all the real manipulable variables needed to operate the plant. Therefore, only a *SISO* controller needs to be adjusted in order to control this new system. The manipulated variable is

$$\xi = w_0 P_{MC} + [w_1 \ w_2 \ w_3 \ w_4 \ w_5 \ w_6] \cdot \mathbf{u} + [w_7 \ w_8 \ w_9 \ w_{10} \ w_{11}] \cdot \mathbf{d} \quad (7.1)$$

Where w_i are the coordinates of the principal component, and the vectors \mathbf{u} , \mathbf{d} are the manipulated variables and measurable disturbances, respectively. Then, having a measure of the disturbances and the current value of the pressure P_{MC} , the virtual manipulated variable ξ can be transformed in the new set of real control actions,

$$\mathbf{u}^T = \frac{[\xi - w_0 P_{MC} - \mathbf{d}^T \mathbf{w}_d] \mathbf{w}_u^T}{\mathbf{w}_u \mathbf{w}_u^T} \quad (7.2)$$

Where $\mathbf{w}_d = [w_7 \ w_8 \ w_9 \ w_{10} \ w_{11}]^T$ and $\mathbf{w}_u = [w_1 \ w_2 \ w_3 \ w_4 \ w_5 \ w_6]^T$

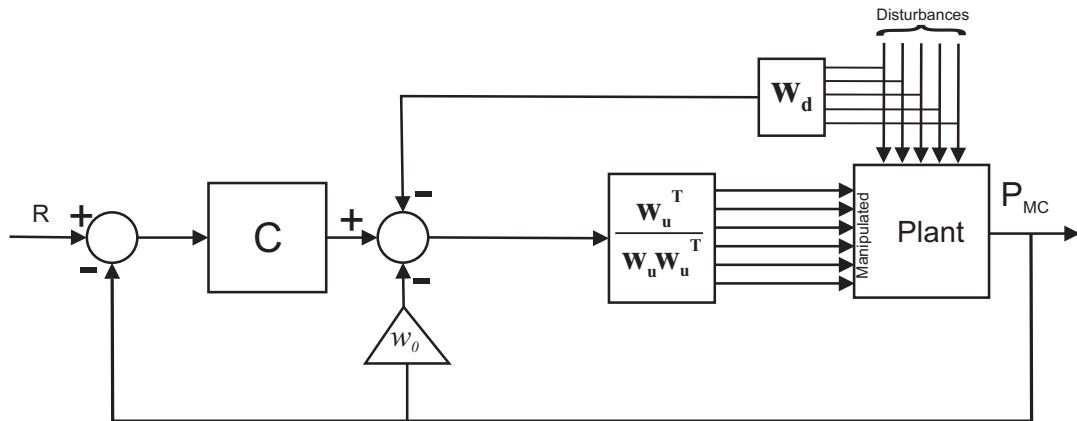


FIGURE 7.1: Control scheme

Tuning a PI controller experimentally over the new *virtual* SISO system and testing in simulation using the neurofuzzy model described in section 3.3.2, using real data from the plant for disturbances, a set point changes are applied to the pressure in the mixing chamber, and the controller is able to follow the new operating point and reject

disturbances, in spite of the nonlinear behavior of the plant and the dimension reduction.

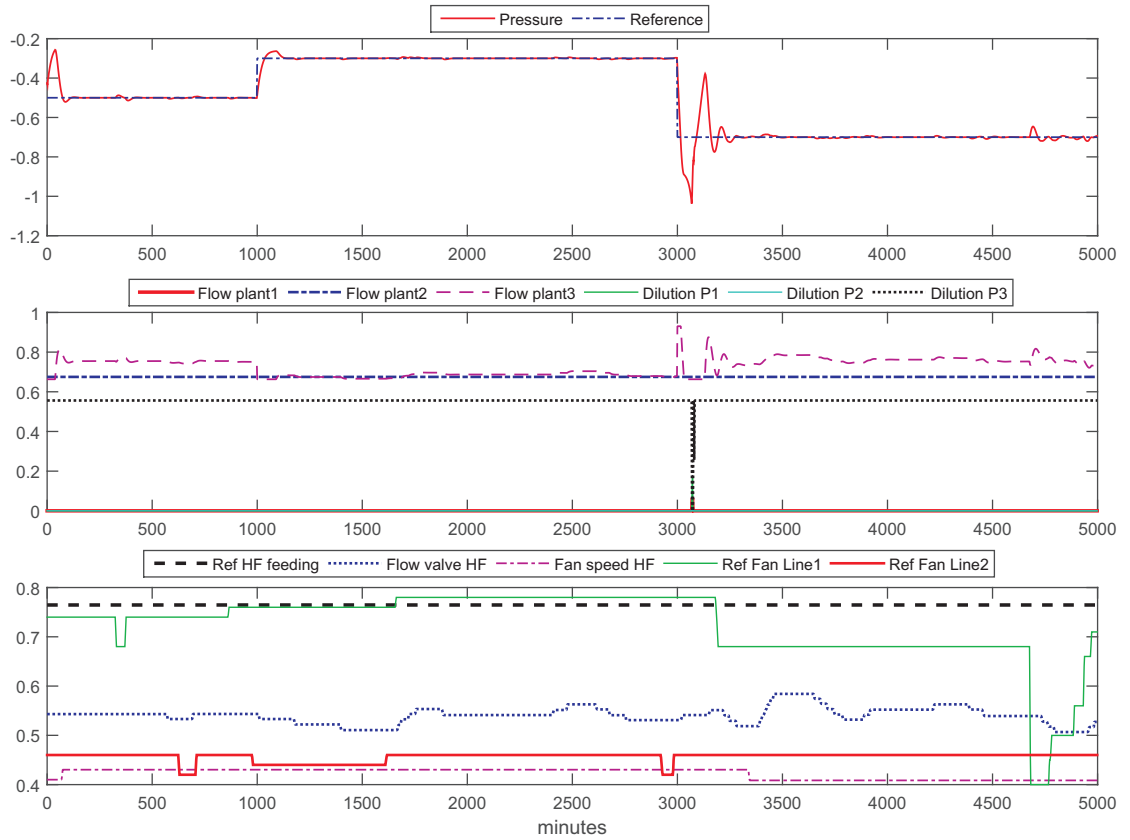


FIGURE 7.2: PI performance for pressure control. Normalised units

Also, using a first order model of the virtual plant obtained after applying the PCA analysis, a model based control strategy can be applied. The resulting model is:

$$G(s) = \frac{-0.49652}{1 + 0.001s} \quad (7.3)$$

In Figure 7.3, the performance of the linear model can be seen, being validated using data from the real process. The input to the model is the new variable ξ while P_{MC} is taken as the output (y). A predictive controller (see sec.1.1) will be designed based on the model above. The sequence of future control signals are calculated such that they minimize a multistage cost function defined by:

$$J(N_1, N_2, N_u) = \sum_{j=N_1}^{N_2} \delta(j) [\hat{y}(t+j | t) - w(t+j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta \xi(t+j-1)]^2 \quad (7.4)$$

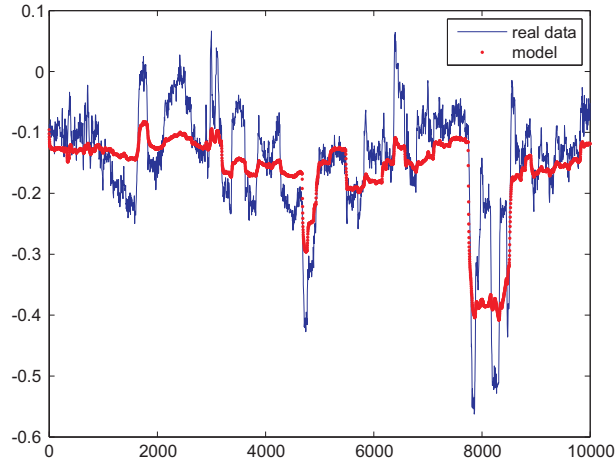


FIGURE 7.3: Performance of the linear model

Where $\hat{y}(t + j | t)$ is a j -step ahead prediction of the system output on data up to time t , N_1 and N_2 are the minimum and maximum prediction horizon, $\delta(j)$ and $\lambda(j)$ are weighted sequences, and $w(t + j)$ is the future reference trajectory.

To solve this problem, as we saw in section 1.1, we use equations 1.10 and followings. The horizon can be defined by $N_1 = d + 1$, $N_2 = d + N$ and $N_u = N$. To solve the GPC problem, the set of control signals $\xi = [\xi(t), \xi(t + 1), \dots, \xi(t + N)]^T$ has to be obtained in order to optimize expression (7.4). As the cost function is quadratic, its optimum can be easily obtained, assuming there are no constraints on the control signals, by making the gradient of J equal to zero. Considering $\delta(j)$ and $\lambda(j)$ are constants and grouping the terms of equation (1.13) which depend on the past, into \mathbf{f} , it leads to:

$$\xi = (G^T G + \lambda I)^{-1} G^T (\mathbf{w} - \mathbf{f}) \tag{7.5}$$

where $G = [g_{d,0} \ g_{d+1,0} \ \dots \ g_{N_p,0}]^T$ and $\mathbf{w} = [w(t + d + 1) \ w(t + d + 2) \ \dots \ w(t + d + N)]^T$. The control signal that is sent to the process is the first element of ξ , given by:

$$\Delta \xi(t) = \mathbf{K}(\mathbf{w} - \mathbf{f}) \tag{7.6}$$

The GPC tuning parameter (move suppression λ , has been adjusted experimentally, and the controller have been tested in simulation as the previous PI, using real data from the plant for disturbances. The controller performance can be seen in figure Both controllers show similar performance. This analysis helps, through a modelling simplification technique, to simplify the control scheme.

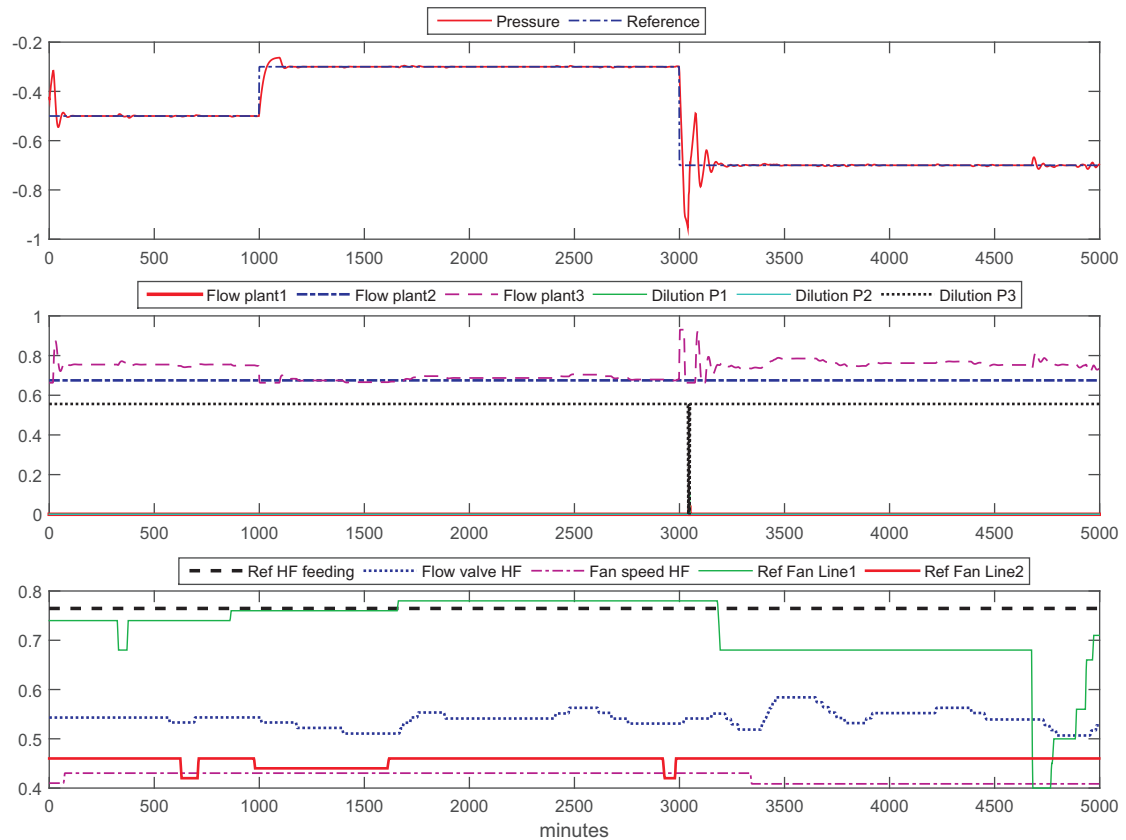


FIGURE 7.4: GPC performance for pressure control. Normalised units

7.2 Application of FPCA to FMPC without constraints

In section 6.5.2 an example of a mechanical system was presented. Following the same procedure presented in section 5.1.2, for each of the consequents of the fuzzy system used for the model, a linear GPC controller can be designed. The advantage of this technique is the simplistic natural way of translating the GPC (or DMC) to linear spaces (consequent of each rule). In this particular example, only 9 controllers must be designed. However, the problem arises when the number of rules increases. The complexity reduction technique shown in chapter 6, can overcome this problem in an efficient manner. Expression 6.37 shows the principal component containing the maximum variability and 6.36 the combination of the new consequent and the principal component. Based on the new consequent, just one GPC design is required. A comparison between three control strategies will be carry out over the mechanical system. The first is a classical PID, adjusted to work around an operation point, the second is a linear GPC designed over the same point, based on a linear model and the third is a Fuzzy GPC with the reduction of complexity produced by FPCA in the model (6.5.2). Figure

7.5 shows a regular performance, independent of the operating point is observed for the FGPC. This scheme can be seen as a linear controller $u_L(k)$ (consequent) modulated by a nonlinear factor $\psi(k)$ (antecedent).

$$u(k) = \psi(k)u_L(k) \quad (7.7)$$

The controller $u_L(k)$ is designed using 6.17, and applying FPCA,

$$\tilde{\mathbf{g}}(x) = \mathbf{G} \cdot \mathbf{a}(x) = \mathbf{H} \cdot \xi(x) \quad (7.8)$$

Where \mathbf{H} is the new consequent and $\xi(x)$ the antecedent. Knowing 6.21,

$$\begin{aligned} \mathbf{G} \cdot \mathbf{a}(x) &= \mathbf{H} \cdot \mathbf{a}(x) \cdot \mathbf{b}^T \\ \mathbf{G} \cdot \mathbf{a}(x) \cdot \mathbf{b}^T &= \mathbf{H} \cdot \mathbf{a}(x) \cdot \mathbf{b}^T \cdot \mathbf{b}^T \\ \mathbf{G} &= \mathbf{H} \cdot \mathbf{b}^T \\ \mathbf{H} &= \frac{\mathbf{G} \cdot \mathbf{b}}{\mathbf{b}^T \mathbf{b}} \end{aligned} \quad (7.9)$$

$\frac{1}{\mathbf{b}^T \mathbf{b}}$ can be written as $\frac{1}{\mathbf{b}^T \mathbf{b}} = \varepsilon \cdot \eta$, having:

$$\tilde{\mathbf{g}}(x) = \mathbf{H} \cdot \xi(x) = \eta \cdot \mathbf{G} \cdot \mathbf{b} \cdot \varepsilon \cdot \xi(x) \quad (7.10)$$

using $\eta \cdot \mathbf{G} \cdot \mathbf{b}$ as a linear model to design a GPC and modulating the nonlinear term $\varepsilon \cdot \xi(x)$, a stable solution can be found as Figure 7.5 shows.

7.3 FPCA applied to PWA systems

As discussed in section 1.3, an implementation of MPC in low-capability hardware is getting an explicit Optimizer solution [52–55]. Using mpQP a controller is defined as a PWA, generally formulated as:

```

IF  $\mathbf{x} \in \Theta_1$  THEN  $u = f_1(\mathbf{x})$ 
ELSIF  $\mathbf{x} \in \Theta_2$  THEN  $u = f_2(\mathbf{x})$ 
:
ELSIF  $\mathbf{x} \in \Theta_N$  THEN  $u = f_N(\mathbf{x})$ 
ENDIF

```

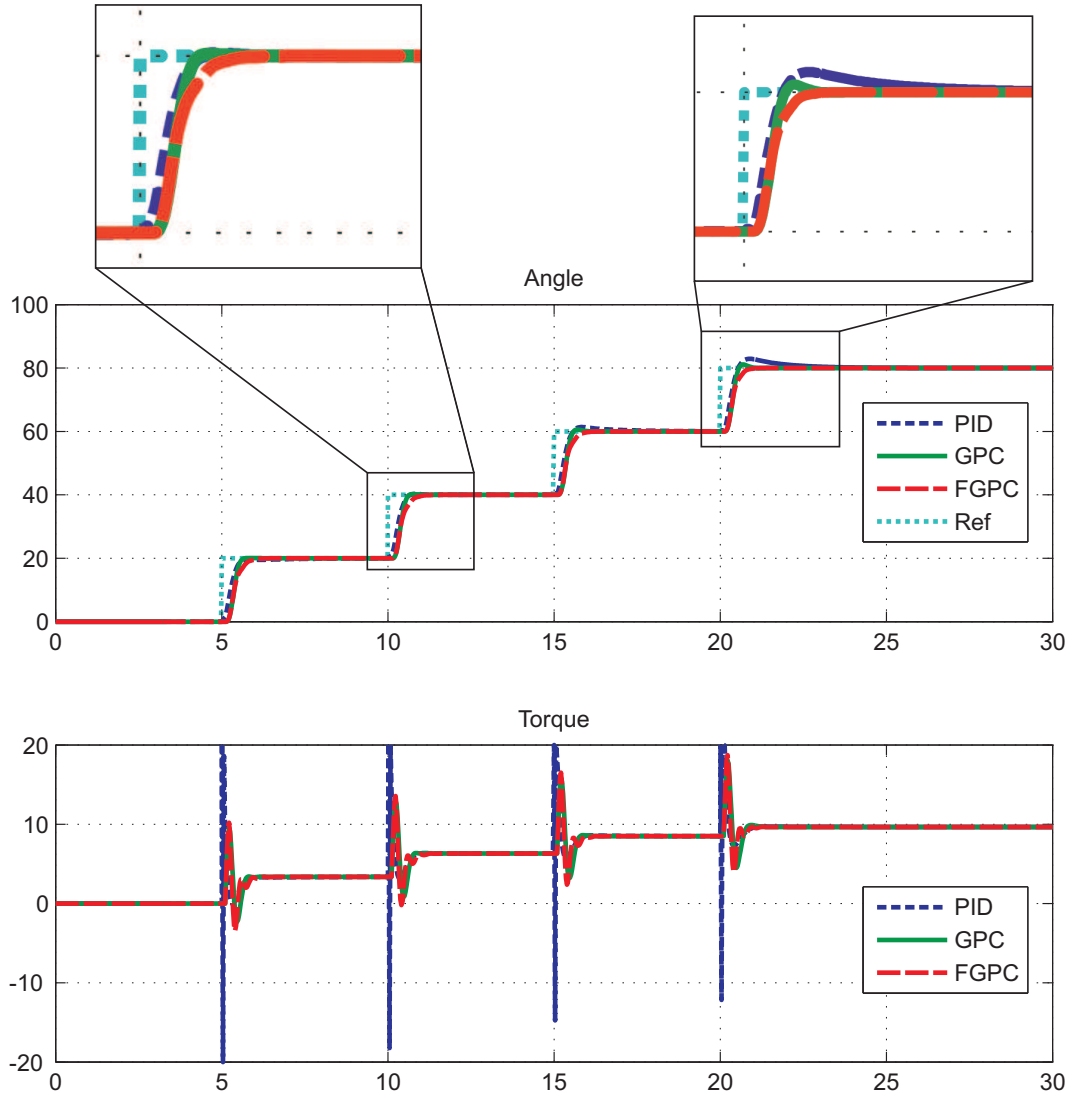


FIGURE 7.5: Mechanical system position control comparison

Each region Θ_i is defined by a polythope. Let δ_i be functions defined as:

$$\delta_i(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Theta_i \\ 0 & \text{else} \end{cases} \quad (7.11)$$

If N is the number of regions, the piece wise affine system can be expressed by:

$$u = \sum_{i=1}^N \delta_i(\mathbf{x}) f_i(\mathbf{x}) \quad (7.12)$$

and $\sum_{i=1}^N \delta_i = 1$.

Obtaining a PWA system as a MPC controller solves the problem of real-time computation of MPC optimisation, but if the PWA needs a large number of regions to obtain a good solution, the programming effort and memory capacity of the hardware platform,

can cause the non-applicability of this methodology, preventing predictive control for fast and / or low cost systems. There have been studies in this field, where methods reduce the number of regions of PWA, allowing for some loss of optimality. [290] derives a exploration strategy for subdividing the parameter space, which avoids unnecessary partitioning, [291] presents a technique by relaxing the Karush-Kuhn-Tucker (KKT) conditions for optimality. A rotation of the state space, obtaining a suboptimal control action is presented in [292]. In [293], T. Johansen proposes a method to reduce complexity, if in the formulation of the optimisation problem:

$$\begin{aligned} J(U, x) &= \frac{1}{2}U^T U + x^T F U + \frac{1}{2}x^T Y x \\ \text{s.t.} \quad &GU \leq W + Ex \end{aligned} \quad (7.13)$$

A SVD analysis is applied in order to replace the terms Ex and $F^T x$ with approximated linear terms defined on a subspace of the state space. Another approach for reducing complexity will be presented in this thesis. Following the formulation presented in Chapter 6 and developing equation 7.12 as:

$$u = \sum_{i=1}^N \delta_i(\mathbf{x})(\rho_{0i} + \rho_{1i}x_1 + \dots + \rho_{ni}x_n) \quad (7.14)$$

Where n is the number of inputs. Thus, the expression 7.14 may take a form similar to 6.17:

$$u(\mathbf{x}) = \tilde{g}_0(\mathbf{x}) + \tilde{g}_1(\mathbf{x})x_1 + \dots + \tilde{g}_n(\mathbf{x})x_n \quad (7.15)$$

Where:

$$\tilde{g}_i(\mathbf{x}) = \sum_{j=1}^N \delta_j(\mathbf{x}) \cdot \rho_{ji} \quad (7.16)$$

And for all the regions $\tilde{\mathbf{g}}$ is:

$$\tilde{\mathbf{g}}(\mathbf{x}) = \begin{bmatrix} \tilde{g}_0(\mathbf{x}) \\ \tilde{g}_1(\mathbf{x}) \\ \vdots \\ \tilde{g}_n(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \rho_{10} & \rho_{20} & \dots & \rho_{N0} \\ \rho_{11} & \rho_{21} & \dots & \rho_{N1} \\ & \vdots & & \\ \rho_{1n} & \rho_{2n} & \dots & \rho_{Nn} \end{bmatrix} \cdot \begin{bmatrix} \delta_0(\mathbf{x}) \\ \delta_1(\mathbf{x}) \\ \vdots \\ \delta_N(\mathbf{x}) \end{bmatrix} \quad (7.17)$$

$$\tilde{\mathbf{g}}(x) = \mathbf{R} \cdot \Delta(x) \quad (7.18)$$

The covariance functions of $\tilde{\mathbf{g}}(x)$, are:

$$\text{Cov}[\tilde{\mathbf{g}}(\mathbf{x}), \tilde{\mathbf{g}}(\mathbf{s})] = \Delta(\mathbf{x})^T \text{cov}(\mathbf{R}) \Delta(\mathbf{s}) \quad (7.19)$$

Supposing that the eigen functions are:

$$\Gamma(x) = \begin{bmatrix} \gamma_1(x) \\ \gamma_2(x) \\ \vdots \\ \gamma_h(x) \end{bmatrix} = \Delta(x)^T \cdot \mathbf{b} \quad (7.20)$$

Thus, taking in account (7.19):

$$\begin{aligned} \int_0^X \text{Cov}[\tilde{\mathbf{g}}(x), \tilde{\mathbf{g}}(s)] \cdot \gamma(s) ds &= \int_0^X \Delta(x)^T \text{cov}(\mathbf{R}) \Delta(s) \cdot \Delta(s)^T \cdot \mathbf{b} ds \\ &= \Delta(x)^T \text{cov}(\mathbf{R}) \cdot \mathbf{W} \cdot \mathbf{b} \\ \text{cov}(\mathbf{R}) \cdot \mathbf{W} \cdot \mathbf{b} &= \lambda \cdot \mathbf{b} \end{aligned} \quad (7.21)$$

Where:

$$\mathbf{W} = \int_0^X \Delta(s) \cdot \Delta(s)^T ds \quad (7.22)$$

Being $\gamma(x)$ orthogonal, then $\langle \gamma_i(x), \gamma_j(x) \rangle = b_i^T \cdot \mathbf{W} \cdot b_j = 0$. As it was seen in 6.24:

$$\mathbf{W}^{\frac{1}{2}} \cdot \text{cov}(\mathbf{R}) \cdot \mathbf{W}^{\frac{1}{2}} \cdot \mathbf{p} = \lambda \cdot \mathbf{p} \quad (7.23)$$

A symmetric eigenvalue problem now remains to be solved. Since only one region will be active in each state vector, equation 7.20 will give only one of the values within \mathbf{b} parameter,

$$\gamma_i(x) \in \{b_{i1}, b_{i2}, \dots, b_{iN}\} \quad (7.24)$$

It may be that there are repeated b_{ij} values, in such case, the regions associated to those values can be merged, reducing the number of regions, i.e. If $b_{ij} = b_{ik}$ with $j \neq k$, then $\gamma_i(x) = b_{i1} \delta_1(\mathbf{x}) + \dots + b_{ij} (\delta_i(\mathbf{x}) + \delta_j(\mathbf{x})) + \dots + b_{iN} \delta_N(\mathbf{x})$, being:

$$\delta_i(\mathbf{x}) + \delta_j(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Theta_i \cup \Theta_j \\ 0 & \text{else} \end{cases} \quad (7.25)$$

7.3.1 Example: distillation column

To illustrate the performance of a reduced PWA system, a high purity distillation column, like the example in section 4.3.1 will be given. The distillation process typically works around an operating point, being identifiable by a linear model. The example will be carried out using the model shown in [294], where the linear model in an operational point is defined as:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -0.0133 & 0 \\ 0 & -0.0133 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0.0117 & 0.0115 \\ 0.0144 & 0.0146 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (7.26)$$

Where y_1 and y_2 are the top and bottom product compositions, respectively, and the inputs, u_1 and u_2 , are the reflux flow rate and the boil-up, respectively, as is shown in the Figure 7.6 Considering the reference tracking problem, i.e., the problem of driving

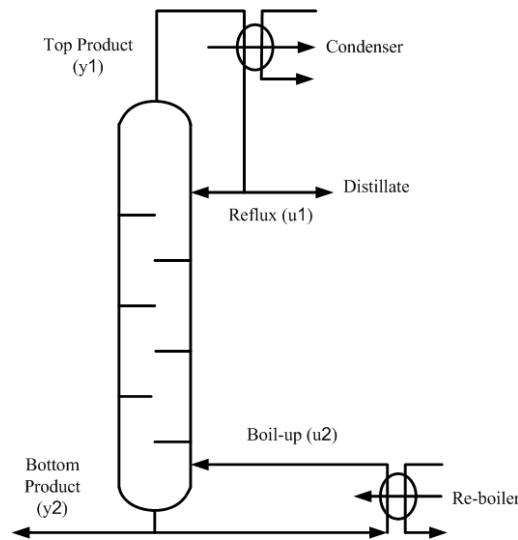


FIGURE 7.6: Distillation Column [295]

the output (Product composition) y to track a given reference signal $r \in \mathcal{R}^p$ by adjusting the control inputs (reflux and boiler flow rates) u under the control input and control increment constraints. For the current x , the constrained MPC solves the following optimisation problem such that the optimal control increment Δu is found at each sampling

instant,

$$\begin{aligned}
& \min_{\mathbf{u}} \left\{ \mathcal{J}(\mathbf{u}, \mathbf{r}, y(k)) \right\} \\
& \text{s. t. } u_{\min} \leq u(k+j|k) \leq u_{\max}, \quad j = 0, \dots, N_u - 1, \\
& \quad x(k+j+1|k) = Ax(k+j|k) + Bu(k+j), \quad j \geq 0, \\
& \quad y(k+j|k) = Cx(k+j|k) + Du(k+j), \quad j \geq 0, \\
& \quad u(k+j) = u(k+j-1) + \Delta u(k+j), \quad j \geq 0,
\end{aligned} \tag{7.27}$$

where the cost function to be minimised is given by:

$$\begin{aligned}
\mathcal{J}(\mathbf{u}, \mathbf{r}, y(k)) &= \sum_{j=0}^{N_y-1} [y(k+j|k) - r(k)]^T Q \\
& \quad [y(k+j|k) - r(k)] \\
& \quad + \sum_{j=0}^{N_u-1} \Delta u(k+j)^T R \Delta u(k+j),
\end{aligned} \tag{7.28}$$

and $\mathbf{u} \triangleq [\Delta u(k)^T, \dots, \Delta u(k+N_u-1)^T]^T$, $x(k+j|k)$ is the predicted state at time step k , N_y and N_u are the prediction and control horizons, and $Q \geq 0$, $R > 0$.

The above MPC optimisation problem (7.27) can be described in the standard QP form, [18], and as shown in [53] such an MPC QP problem can be transformed into:

$$\begin{aligned}
& \min_{\mathbf{u}} \left\{ \mathcal{J}(\mathbf{u}, \theta(k)) = \frac{1}{2} \mathbf{u}^T H \mathbf{u} + \theta(k)^T F^T \mathbf{u} \right\} \\
& \text{s. t. } \quad G \mathbf{u} \leq W + S \theta(k),
\end{aligned} \tag{7.29}$$

where \mathbf{u} is defined as in (7.28), and θ is the vector of parameters defined as:

$$\theta(k) = [\mathbf{x}(k), \mathbf{u}(k-1), \mathbf{r}(k)]^T$$

The MPC mpQP problem (7.29) is solved explicitly, off-line, for *all* the feasible values of θ of interest, resulting in the solution $\mathbf{u}^*(\theta)$, which is a continuous piecewise-affine function defined over a polyhedral partition in the θ -space represented as:

$$\Delta u(k) = f(\theta(k)) = \begin{cases} K_1 \theta(k) + k_1, & \text{if } \theta(k) \in \Theta_1 \\ K_2 \theta(k) + k_2, & \text{if } \theta(k) \in \Theta_2 \\ \vdots \\ K_{N_{rej}} \theta(k) + k_{N_{rej}}, & \text{if } \theta(k) \in \Theta_{N_{rej}}, \end{cases} \tag{7.30}$$

with a polyhedral partition $\mathcal{P} = \{\Theta_1, \dots, \Theta_{N_{rej}}\}$, where the polyhedral sets are represented by linear inequalities (hyperplanes),

$$\Theta_i = \{\theta(k) | L_i \theta(k) \leq l_i\}, i = 1, \dots, N_{rej}. \quad (7.31)$$

Here, K_i and k_i are the control gain and offset for each region respectively, and N_{rej} is the number of regions. Consequently, the on-line temperature control algorithm is reduced to a look-up table: the region associated with the current state θ is first determined, and then the optimal control law valid for that region is applied.

The tuning parameters used for deriving the explicit MPC controller are as follows: $N_y = 20$, $N_u = 3$, $\mathbf{Q} = \mathbf{R} = \mathbf{I}$, with a sampling time of 10 min. The control input constraints are given as $-2 \leq u \leq 2$ and $-1 \leq u \leq 2$, respectively. The number of regions obtained for the control law is 140. Applying the FPCA above to this application, with the same tuning, just 5 consequents are obtained for the first manipulate variable and 6 for the second,

$$\tilde{g}_1(\mathbf{x}) = \begin{pmatrix} 0.0177 & -0.1947 & 0.1562 & 0.0036 & 0.0096 \\ -0.0207 & 0.2275 & -0.1822 & 0.0075 & -0.0103 \\ -0.1298 & 0.8182 & 0.4014 & -0.0068 & -0.0012 \\ -0.0003 & 0.0035 & -0.0027 & -0.0007 & -0.0004 \\ -0.0302 & 0.3351 & -0.2730 & -0.0245 & 0.0031 \\ 0.0336 & -0.3704 & 0.3005 & -0.0195 & -0.0085 \\ 1.6627 & 0.2191 & 0.0795 & -0.0067 & -0.0013 \end{pmatrix} \cdot \Delta(\mathbf{x})$$

$$\tilde{g}_2(\mathbf{x}) = \begin{pmatrix} -0.0271 & 0.1330 & -0.1683 & -0.0030 & 0.0074 & 0.0045 \\ -0.0322 & 0.1575 & -0.1992 & 0.0083 & 0.0092 & 0.0027 \\ -0.0006 & 0.0027 & -0.0033 & -0.0029 & -0.0003 & 0.0032 \\ -0.4149 & 0.8620 & 0.2987 & 0.0018 & 0.0062 & 0.0035 \\ 0.0455 & -0.2331 & 0.2968 & 0.0203 & 0.0060 & 0.0038 \\ 0.0504 & -0.2562 & 0.3258 & -0.0136 & 0.0088 & 0.0033 \\ 1.3315 & 0.4020 & 0.1321 & 0.0018 & 0.0062 & 0.0035 \end{pmatrix} \cdot \Delta(\mathbf{x})$$

Associating regions with the same parameter as it has been seen in 7.25, the arithmetic operations of antecedents are reduced from 140 to 68. In Figure 7.7 it can be observed that the proposed transformation has a similar performance as the system.

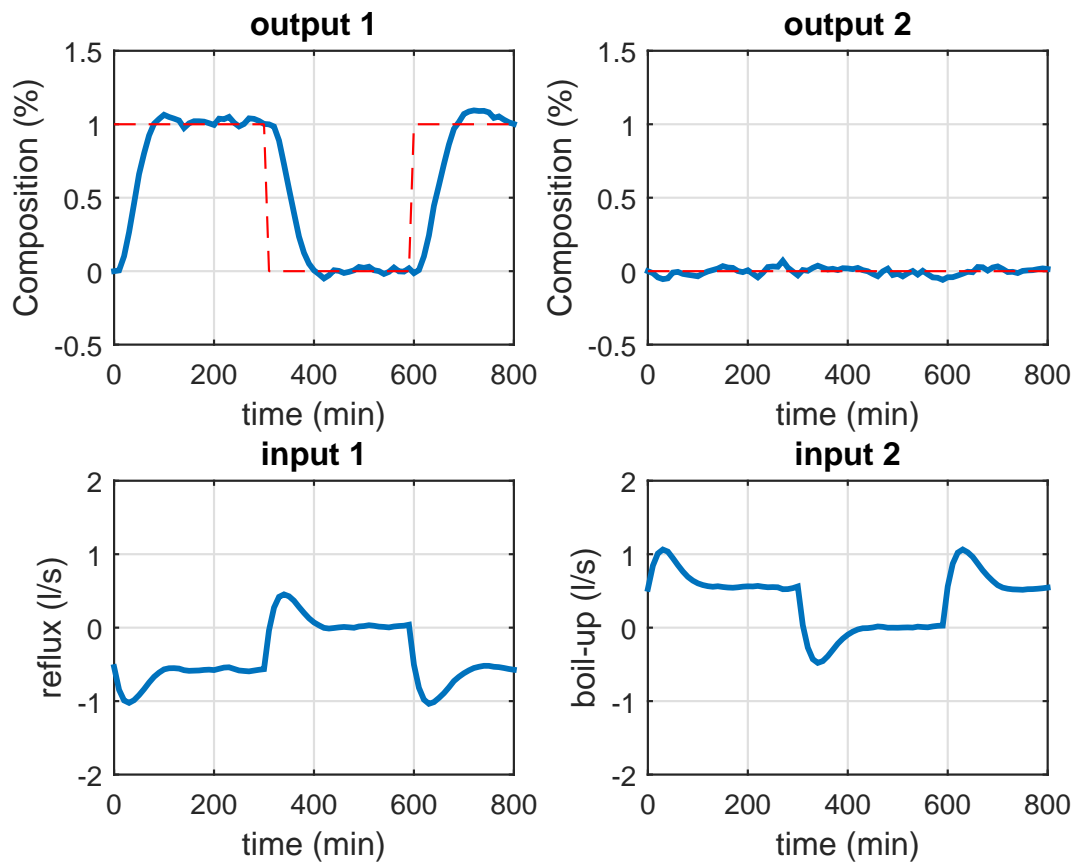


FIGURE 7.7: Controllers performance for distillation column

7.4 Simplified FMPC with constraints

The idea indicated in 1.4 about the lack of parameter setting in the MPC with explicit solution and constraints, can be addressed by the application of a fuzzy controller having the control actions of the various PWA designed for different parameter values as inputs, and setting the parameter itself. Figure 7.8 shows the scheme of this structure. This scheme requires more memory capacity and programming time. If greater accu-

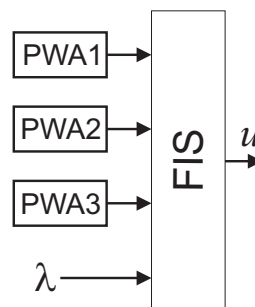


FIGURE 7.8: FMPC with explicit solution subject to constraints

racy is desired, there should be more PWA controller programmed into the system. For example, in the previous application, three different PWA for three values of λ are designed, giving the number of regions and the performance shown in the figure 7.9. Once the FPCA is applied, a huge reduction in the number of consequents can be

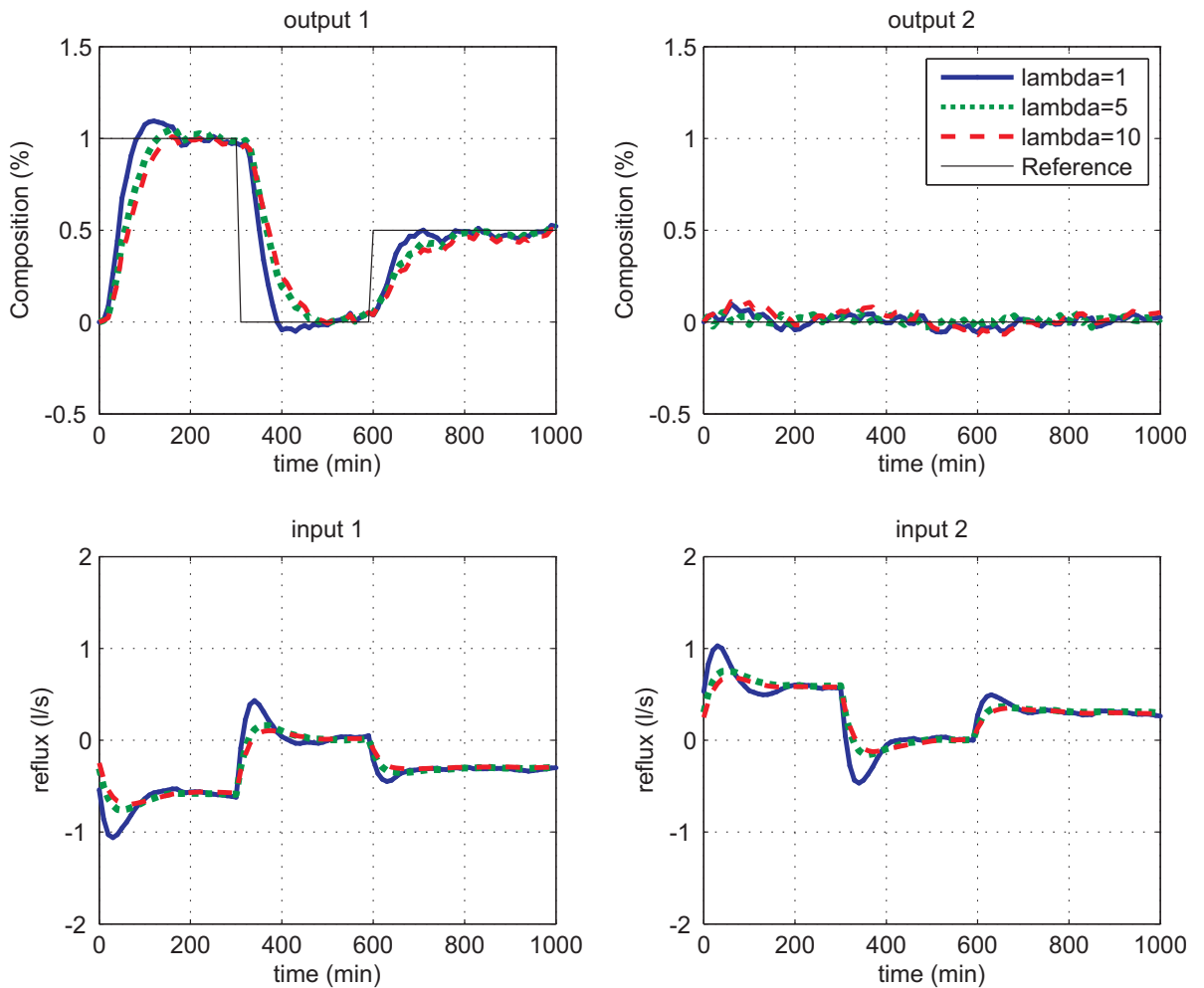


FIGURE 7.9: PWA controllers depending on λ

observed (see table 7.1), and following 7.25, an association of regions can be done, reducing the antecedents between 45 and 51%. Following the structure of the system proposed above (see Figure 7.8), the parameter λ (move suppression) can be changed in order to give more or less aggressiveness to the system. Figure 7.10 shows the performance of the controller when λ takes different values between those which are in the table 7.1.

TABLE 7.1: Controllers structure using FPCA

| <i>parameter</i> | <i>Variable</i> | <i>Rules number</i> | <i>Reduced consequents number/ reduction(%)</i> | <i>Reduced antecedents number/ reduction(%)</i> |
|------------------|-----------------|---------------------|---|---|
| $\lambda = 1$ | u1 | 140 | 5 (96%) | 68 (51%) |
| $\lambda = 1$ | u2 | 140 | 6 (95%) | 71 (49%) |
| $\lambda = 5$ | u1 | 162 | 7 (95.6%) | 89 (45%) |
| $\lambda = 5$ | u2 | 162 | 5 (97%) | 87 (46%) |
| $\lambda = 10$ | u1 | 161 | 5 (97%) | 88 (45%) |
| $\lambda = 10$ | u2 | 161 | 6 (96%) | 87 (46%) |

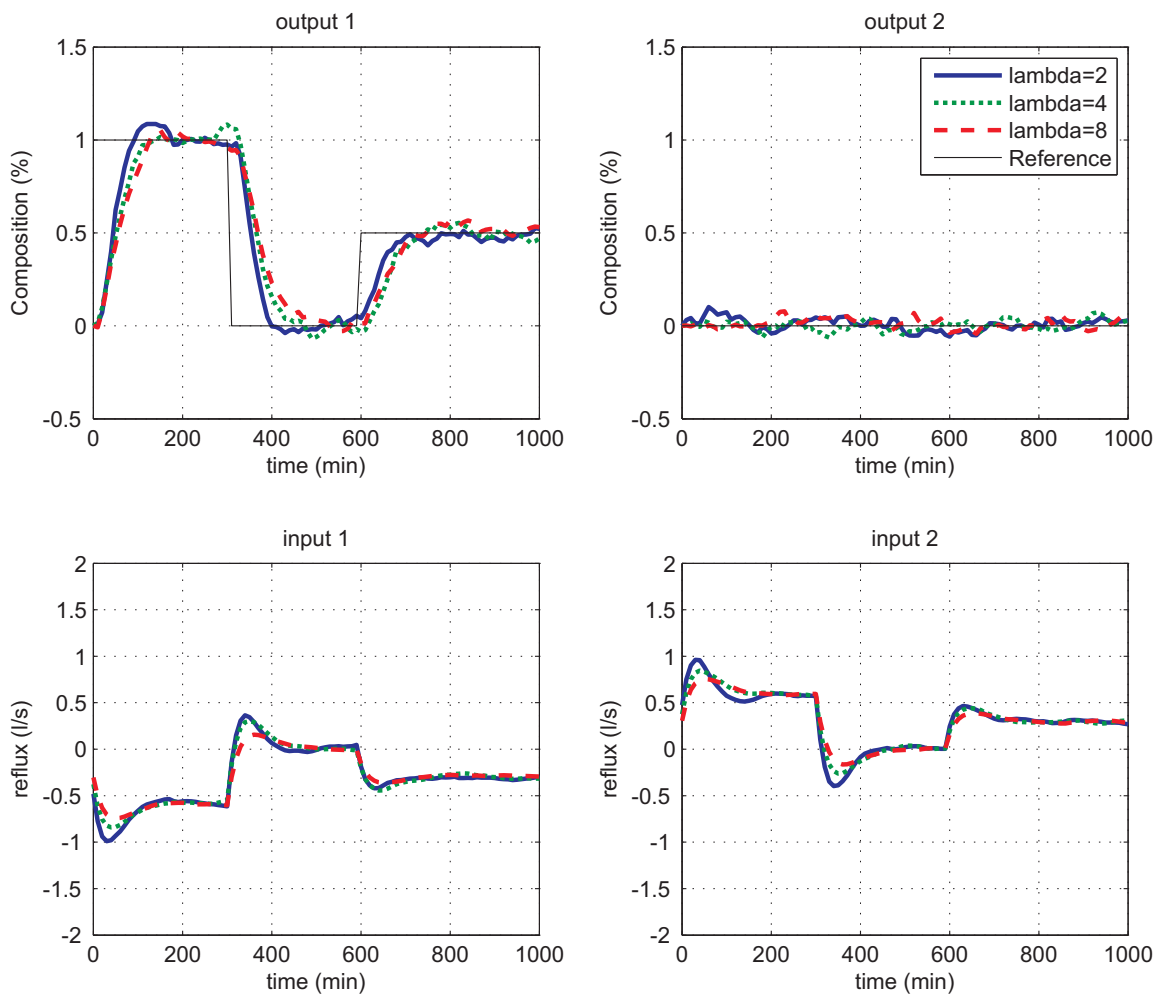


FIGURE 7.10: FMPC for distillation column depending on lambda

7.5 Conclusion to the chapter

In this chapter three methods to reduce complexity have been seen. Firstly, reducing the dimension of the space of manipulated and disturbance variables to one, has permitted to reduce a MISO system to SISO. The second way is the application of the technique studied in chapter 6, reducing the number of rules of a fuzzy system and designing GPC controllers from the new consequents. An example has been a position control of a mechanism, reducing complexity just to one rule, designing one GPC to control the nonlinear system. The third way is the application of the FPCA to a PWA system to reduce its complexity. This last way permits to reduce complexity of the explicit solution of the optimisation problem subject to constraints, using an adjusting parameter. In order to use MPC with constraints on a system, we have to consider several issues: The computational load, which is determined by the number of operations performed. The use of on-line optimisation can become limited by the speed of the microprocessor in computing with a high number of variables. The explicit calculation of the optimisation solution solves this problem but a memory shortage may sometime limit its implementation, especially when is necessary to have a fitting parameter. Furthermore, in the actual implementation of a system, the time spent in programming, which is not negligible, must be taken into account, considering the lack of tools to automatically generate the appropriate code from the design and choice of industrial PLC and embedded systems. In this chapter a combination of the reduction technique showed in Chapter 6 applied on PWA systems and, together with the use of a fuzzy system, has succeeded in designing a MPC scheme with constraints and adjustment parameter, with drastically reducing the number of regions of consequents, improving programming time (down 96%) without excessive increase in the computational load. The technique applied to the PWA also allows a new merge of regions (not necessarily adjacent), opening the door to future research on reducing their complexity.

Conclusion and future work

In this chapter, some very general concluding remarks and future work is presented. This thesis has focused on developing a new methodology for complexity reduction in rule based systems.

The main contribution of the thesis is in Chapter 6, i.e. the application of *Functional Principal Components Analysis* to the Hilbert space of functions defined in a rule base. Defining a covariance operator, the technique permits the reduction of the space to a subspace where the operator's eigenvalues are bigger. It has been published in [296].

A second contribution, shown in Chapter 7, is the application of the technique described in Chapter 6, to permit the structure reduction of the controller internally, either reducing the number of rules of a fuzzy system and designing GPC controllers from the new consequents, or reducing the rules of a PWA controllers which are the application of model predictive control, subject to constraints with explicit solution. As examples, a position control of a mechanism, reducing complexity just to one rule, designing one GPC to control the nonlinear system and a high distillation column, applying a linear GPC subject to constraints, with explicit solution of the optimisation problem, using a tuning parameter.

Other derived contribution from the PCA application the reduction of a MISO system to SISO, which simplifies the complexity of the control system, by reducing the dimension of the space of the manipulated and disturbance variables to one, as is shown in section 7.1.

Many practical applications of MPC implementation in low capability hardware and fuzzy systems which can be viewed as minor contributions of this thesis. The general fuzzy modelling methodology requires a considerable amount of empirical knowledge and experience [297]. The practical implementation of MPC and fuzzy systems, has resulted in several publications and the acquisition of experience needed to model and control systems in industrially feasible manner.

A practical contribution of RSSI-based explicit GPC and a min-max MPC approach has

been presented to address the radio power control problem encountered in ambulatory sensor networks. The results have been published already in [60], [61] and [62]. It has been shown that an explicit solution of the constrained min-max MPC problem can be computed for the WSN power control problem by solving an mpQP. The feasibility of the proposed design and its performance has been experimentally validated.

A basic real contribution of a FIS has been a real software application for commercial purposes using fuzzy techniques. The application belongs to the company Sensipass[®] and two articles have been published in [102] and [103]. A contribution of fuzzy modelling been presented in section 3.2.3, applying GA to chose the input field structure for a fuzzy model, using the Mackey-Glass chaotic series as an example of prediction. Others contributions have been several real application of modelling published in [298] and [?] for an industrial autoclave and [169], [299] and [300] for an industrial gas mixing chamber. An application of a supervisory fuzzy control system in a real chemical plant in operation has been applied. A comparison of other strategies with a direct fuzzy controller in an air levitation plant has been made and published in [301] and [180]. In Chapter 5, two different schemes have been implemented in a real industrial plant, using low computational cost hardware with FCL. The results have been compared with a linear GPC, obtaining better performance with no significant programming effort and computational resources. The results have been published in [298] and [?].

Future works

The application of the reduction technique presented in Chapter 6 may result in a lack of interpretability in the fuzzy system, due to the algebraic combination before aggregation. If we focus on the use in predictive control, this is not a problem. Even predictive controllers with explicit solutions that could be implemented in industrial devices according to IEC61131-3 [7] and IEC61131-7 [11], there may be the possibility of non-prototypical membership functions or algebraic operations after fuzzification. Therefore, a future work may be to focus to obtain interpretable antecedents using a proper transformation of the input space, getting a simpler way to implement the reduction technique.

The technique applied to the PWA also allows a new merge of regions (not necessarily adjacent), opening the door to future research on reducing their complexity when an explosion of rules arises due the accuracy of modelling.

Bibliography

- [1] Yucai Zhu. *Multivariable system identification for process control*. Elsevier, 2001.
- [2] Zhiqiang Gao and R Russell Rhinehart. Theory vs. practice: The challenges from industry. In *Proc. of American Control Conference*, pages 1341–1349, 2004.
- [3] C. Bordons, M.R. Arahal, E.F. Camacho, and J.M. Tejera. Energy saving in a copper smelter by means of model predictive control. In Ricardo S. Sánchez Peña, Vicenç Puig Cayuela, and Joseba Quevedo Casín, editors, *Identification and Control*, pages 63–85. Springer London, 2007. ISBN 978-1-84628-898-2. doi: 10.1007/978-1-84628-899-9_3. URL http://dx.doi.org/10.1007/978-1-84628-899-9_3.
- [4] Daniele Pugliesi. *Functional levels of a Distributed Control System*. Licensed under CC BY-SA 3.0. URL http://commons.wikimedia.org/wiki/File:Functional_levels_of_a_Distributed_Control_System.svg.
- [5] CENELEC. *Programmable controllers - Part 1: General information, ed 2.0*. CENELEC, 2003.
- [6] CENELEC. *Programmable controllers - Part 2: Equipment requirements and tests, ed 2.0*. CENELEC, 2007.
- [7] CENELEC. *Programmable controllers - Part 3: Programming languages, ed 3.0*. CENELEC, 2013.
- [8] CENELEC. *Programmable controllers - Part 4: User guidelines, ed 2.0*. CENELEC, 2004.
- [9] CENELEC. *Programmable controllers - Part 5: Communications, ed 1.0*. CENELEC, 2000.

- [10] CENELEC. *Programmable controllers - Part 6: Functional safety. ed 1.0*. CENELEC, 2012.
- [11] CENELEC. *Programmable controllers - Part 7: Fuzzy Control programming. ed 1.0*. CENELEC, 2000.
- [12] CENELEC. *Programmable controllers - Part 8: Guidelines for the application and implementation of programming languages. ed 2.0*. CENELEC, 2003.
- [13] CENELEC. *Programmable controllers - Part 9: Single-drop digital communication interface for small sensors and actuators (SDCI). ed 1.0*. CENELEC, 2013.
- [14] CENELEC. *Function blocks -Part 1: Architecture. ed 2.0*. CENELEC, 2012.
- [15] CENELEC. *Function blocks -Part 2: Software tool requirements. ed 2.0*. CENELEC, 2012.
- [16] CENELEC. *Function blocks for industrial-process measurement and control systems - Part 3: Tutorial information. ed 1.0*. CENELEC, 2004.
- [17] CENELEC. *Function blocks - Part 4: Rules for compliance profiles. ed 2.0*. CENELEC, 2013.
- [18] E.F. Camacho and C. Bordons. *Model Predictive Control*. Advanced Textbooks in Control and Signal Processing. ISBN 978-1-85233-694-3.
- [19] S.Joe Qin and Thomas A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7):733 – 764, 2003. ISSN 0967-0661. doi: [http://dx.doi.org/10.1016/S0967-0661\(02\)00186-7](http://dx.doi.org/10.1016/S0967-0661(02)00186-7). URL <http://www.sciencedirect.com/science/article/pii/S0967066102001867>.
- [20] S. Joe Qin and Thomas A. Badgwell. A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764, 2003.
- [21] C. Cutler and B. Ramaker. Dynamic matrix control—a computer control algorithm. In *Automatic Control Conference*, 1980.
- [22] D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control—part i. the basic algorithm. *Automatica*, 23(2):137–148, March 1987. ISSN 0005-1098. doi: 10.1016/0005-1098(87)90087-2. URL [http://dx.doi.org/10.1016/0005-1098\(87\)90087-2](http://dx.doi.org/10.1016/0005-1098(87)90087-2).

- [23] P. Zometa, M. Kogel, T. Faulwasser, and R. Findeisen. Implementation aspects of model predictive control for embedded systems. In *American Control Conference (ACC), 2012*, pages 1205–1210, June 2012. doi: 10.1109/ACC.2012.6315076.
- [24] Yang Wang and S. Boyd. Fast model predictive control using online optimization. *Control Systems Technology, IEEE Transactions on*, 18(2):267–278, March 2010. ISSN 1063-6536. doi: 10.1109/TCST.2009.2017934.
- [25] Juan Luis Jerez, Paul J. Goulart, Stefan Richter, George A. Constantinides, Eric C. Kerrigan, and Manfred Morari. Embedded online optimization for model predictive control at megahertz rates. *CoRR*, abs/1303.1090, 2013. URL <http://arxiv.org/abs/1303.1090>.
- [26] S. Joe Qin and Thomas A. Badgwell. An overview of industrial model predictive control technology. *AIChE Symposium Series*, 93(316):232–256, 1997.
- [27] D. Hrovat, S. Di Cairano, H.E. Tseng, and I.V. Kolmanovsky. The development of model predictive control in automotive industry: A survey. In *Control Applications (CCA), 2012 IEEE International Conference on*, pages 295–302, Oct 2012. doi: 10.1109/CCA.2012.6402735.
- [28] J. Richalet, A. Rault, J.L. Testud, and J. Papon. Model predictive heuristic control: Applications to industrial processes. *Automatica*, 14(5):413 – 428, 1978. ISSN 0005-1098. doi: [http://dx.doi.org/10.1016/0005-1098\(78\)90001-8](http://dx.doi.org/10.1016/0005-1098(78)90001-8). URL <http://www.sciencedirect.com/science/article/pii/0005109878900018>.
- [29] J. Richalet, K.E. Åström, and D. O’Donovan. *Predictive Functional Control: Principles and Industrial Applications*. Advances in Industrial Control. ISBN 9781849968454.
- [30] R. Hodrea, I. Nascu, I. Nascu, R. De Keyser, and H. Vasian. Epsac versus pid control of neuromuscular blockade. In *Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on*, pages 1–6, May 2014. doi: 10.1109/AQTR.2014.6857831.
- [31] H.J. Chizeck, Y. Dong, J.M. Khoury, and R.N. Schmidt. Extended horizon adaptive block predictive controller with an efficient prediction system, February 16 1995. URL <http://www.google.com/patents/WO1995004958A1?cl=en>. WO Patent App. PCT/US1994/008,805.

- [32] Siemens. Process control system pcs 7. advanced process library (v8.0 sp2). Technical Report A5E03709256-03, Siemens, June 2013. URL <https://support.industry.siemens.com/cs/#mdm/80539207?c=55656085131&lc=en-WW>.
- [33] Schneider Electric. Predictive control library v3.0, user manual. Technical Report EIO0000000107, Schneider Electric SA, 35 rue Joseph Monier 92500 Rueil Malmaison - France, January 2009. URL http://download.schneider-electric.com/files?p_File_Id=682793230&p_File_Name=EIO0000000107.pdf.
- [34] Willy Wojsznis, John Gudaz, Terry Blevins, and Ashish Mehta. Practical approach to tuning mpc*. *{ISA} Transactions*, 42(1):149 – 162, 2003. ISSN 0019-0578. doi: [http://dx.doi.org/10.1016/S0019-0578\(07\)60121-9](http://dx.doi.org/10.1016/S0019-0578(07)60121-9). URL <http://www.sciencedirect.com/science/article/pii/S0019057807601219>.
- [35] Rahul Shridhar and Douglas J Cooper. A tuning strategy for unconstrained siso model predictive control. *Industrial & engineering chemistry research*, 36(3): 729–746, 1997.
- [36] R. D. Kokate, L. M. Waghmare, and S. D. Deshmukh. Review of tuning methods of dmc and performance evaluation with pid algorithms on a fopdt model. In *Proceedings of the 2010 International Conference on Advances in Recent Technologies in Communication and Computing*, ARTCOM '10, pages 71–75, Washington, DC, USA, 2010. IEEE Computer Society. ISBN 978-0-7695-4201-0. doi: 10.1109/ARTCom.2010.109. URL <http://dx.doi.org/10.1109/ARTCom.2010.109>.
- [37] Charles Ray Cutler. *Dynamic matrix control: an optimal multivariable control algorithm with constraints*. PhD thesis, Houston, TX, USA, 1983.
- [38] Jacinto L. Marchetti, Duncan A. Mellichamp, and Dale E. Seborg. Predictive control based on discrete convolution models. *Industrial & Engineering Chemistry Process Design and Development*, 22(3):488–495, 1983. doi: 10.1021/i200022a025. URL <http://dx.doi.org/10.1021/i200022a025>.
- [39] Paul R. Maurath, D.E. Seborg, and Duncan A. Mellichamp. Predictive controller design by principal components analysis. In *American Control Conference, 1985*, pages 1059–1065, June 1985.

- [40] P.J. Callaghan. *An Experimental Investigation of Predictive Controller Design by Principal Component Analysis*.
- [41] Paul R. Maurath, Duncan A. Mellichamp, and Dale E. Seborg. Predictive controller design for single-input/single-output (siso) systems. *Industrial & Engineering Chemistry Research*, 27(6):956–963, 1988. doi: 10.1021/ie00078a011. URL <http://dx.doi.org/10.1021/ie00078a011>.
- [42] Carlos E. Garcia and Manfred Morari. Internal model control. a unifying review and some new results. *Industrial & Engineering Chemistry Process Design and Development*, 21(2):308–323, 1982. doi: 10.1021/i200017a016. URL <http://dx.doi.org/10.1021/i200017a016>.
- [43] A.R. McIntosh, S.L. Shah, and D.G. Fisher. Selection of tuning parameters for adaptive generalized predictive control. In *American Control Conference, 1989*, pages 1846–1851, June 1989.
- [44] D.W. Clarke and R. Scattolini. Constrained receding-horizon predictive control. *Control Theory and Applications, IEE Proceedings D*, 138(4):347–354, Jul 1991. ISSN 0143-7054.
- [45] J.B. Rawlings and K.R. Muske. The stability of constrained receding horizon control. *Automatic Control, IEEE Transactions on*, 38(10):1512–1516, Oct 1993. ISSN 0018-9286. doi: 10.1109/9.241565.
- [46] Zhi Q. Zheng and M. Morari. Robust stability of constrained model predictive control. In *American Control Conference, 1993*, pages 379–383, June 1993.
- [47] J.H. Lee and Z.H. Yu. Tuning of model predictive controllers for robust performance. *Computers & Chemical Engineering*, 18(1):15 – 37, 1994. ISSN 0098-1354. doi: [http://dx.doi.org/10.1016/0098-1354\(94\)85020-8](http://dx.doi.org/10.1016/0098-1354(94)85020-8). URL <http://www.sciencedirect.com/science/article/pii/0098135494850208>. An International Journal of Computer Applications in Chemical Engineering.
- [48] A.R. McIntosh, D.G. Fisher, and S.L. Shah. Performance adaptive control: General structure and a case study. *Journal of Process Control*, 2(4):213 – 221, 1992. ISSN 0959-1524. doi: [http://dx.doi.org/10.1016/0959-1524\(92\)80010-U](http://dx.doi.org/10.1016/0959-1524(92)80010-U). URL <http://www.sciencedirect.com/science/article/pii/095915249280010U>.

- [49] Ralph F. Hinde and D.J. Cooper. Using pattern recognition in controller adaptation and performance evaluation. In *American Control Conference, 1993*, pages 74–78, June 1993.
- [50] Ralph F. Hinde and Douglas J. Cooper. Unified excitation and performance diagnostic adaptive control framework. *AIChE Journal*, 41(1):110–121, 1995. doi: 10.1002/aic.690410111. URL <http://dx.doi.org/10.1002/aic.690410111>.
- [51] Jorge L. Garriga and Masoud Soroush. Model predictive control tuning methods: A review. *Industrial & Engineering Chemistry Research*, 49(8):3505–3515, 2010. doi: 10.1021/ie900323c. URL <http://dx.doi.org/10.1021/ie900323c>.
- [52] Alberto Bemporad, Manfred Morari, Vivek Dua, and Efstratios N. Pistikopoulos. The explicit solution of model predictive control via multiparametric quadratic programming, 2000.
- [53] Alberto Bemporad, Manfred Morari, Vivek Dua, and Efstratios N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3 – 20, 2002. ISSN 0005-1098. doi: [http://dx.doi.org/10.1016/S0005-1098\(01\)00174-1](http://dx.doi.org/10.1016/S0005-1098(01)00174-1). URL <http://www.sciencedirect.com/science/article/pii/S0005109801001741>.
- [54] E. N. Pistikopoulos. Perspectives in multiparametric programming and explicit model predictive control. *AIChE Journal*, 55(8):1918–1925, 2009. ISSN 1547-5905. doi: 10.1002/aic.11965. URL <http://dx.doi.org/10.1002/aic.11965>.
- [55] Alessandro Alessio and Alberto Bemporad. A survey on explicit model predictive control. In Lalo Magni, Davide Martino Raimondo, and Frank Allgöwer, editors, *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, pages 345–369. Springer Berlin Heidelberg, 2009. ISBN 978-3-642-01093-4. doi: 10.1007/978-3-642-01094-1_29. URL http://dx.doi.org/10.1007/978-3-642-01094-1_29.
- [56] G.P.C. Weinkeller, J.L.F. Salles, and T.F.B. Filho. Predictive control via multiparametric programming applied to the dynamic model of a robotic wheelchair. In *Robotics Symposium and Latin American Robotics Symposium (SBR-LARS), 2012 Brazilian*, pages 179–184, Oct 2012. doi: 10.1109/SBR-LARS.2012.36.

- [57] Michal Kvasnica Miroslav Fikar Ivan Rauová, Richard Valo. Real-time model predictive control of a fan heater via plc. In *International Conference on Process Control (18th: 2011 Tatranská Lomnica, Slovakia)*, pages 388–393, 2011. ISBN 978-80-227-3517-9.
- [58] K. Witthephanich, J.M. Escaño, A.J. Gallego, and E.F. Camacho. Pressurized watertemperature control of a fresnel collector field-type solar cooling system using explicitmodel predictive control. *Asian Conference on Power and Energy Systems. April 2013, Phuket, Thailand.*, 2013.
- [59] M. Walsh, M. Hayes, and J. Nelson. Robust performance for an energy sensitive wireless body area network - an anti-windup approach. *International Journal of Control*, 82(1):59–73, January 2009. ISSN 0020-7179. doi: 10.1080/00207170801983109. URL <http://dx.doi.org/10.1080/00207170801983109>.
- [60] K. Witthephanich, J.M. Escaño, and M.J. Hayes. Ambulatory wireless sensor network power management using constrained explicit generalised predictive control. *International Journal of Control*, 84(8):1450–1466, 2011. doi: 10.1080/00207179.2011.605909. URL <http://dx.doi.org/10.1080/00207179.2011.605909>.
- [61] K. Witthephanich, J.M. Escaño, and M.J. Hayes. Explicitly constrained generalised predictive control strategies for power management in ambulatory wireless sensor network systems. In *American Control Conference (ACC), 2010*, pages 1856–1861. IEEE, 2010.
- [62] K. Witthephanich, J.M. Escaño, D. Muñoz de la Peña, and M.J. Hayes. A min-max model predictive control approach to robust power management in ambulatory wireless sensor networks. *Systems Journal, IEEE*, 8(4):1060–1073, Dec 2014. ISSN 1932-8184. doi: 10.1109/JSYST.2013.2271388.
- [63] Sarangapani. *Wireless Ad Hoc and Sensor Networks Protocols Performance and Control*. CRC Press, 2007.
- [64] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Comput. Netw.*, 52(12):2292–2330, August 2008. ISSN 1389-1286. doi: 10.1016/j.comnet.2008.04.002. URL <http://dx.doi.org/10.1016/j.comnet.2008.04.002>.

- [65] Gang Zhou, Tian He, Sudha Krishnamurthy, and John A. Stankovic. Impact of radio irregularity on wireless sensor networks. In *Proceedings of the 2Nd International Conference on Mobile Systems, Applications, and Services, MobiSys '04*, pages 125–138, New York, NY, USA, 2004. ACM. ISBN 1-58113-793-1. doi: 10.1145/990064.990081. URL <http://doi.acm.org/10.1145/990064.990081>.
- [66] Wei Zhuang, Xi Chen, Jindong Tan, and Aiguo Song. An empirical analysis for evaluating the link quality of robotic sensor networks. In *Wireless Communications Signal Processing, 2009. WCSP 2009. International Conference on*, pages 1–5, Nov 2009. doi: 10.1109/WCSP.2009.5371588.
- [67] Guang-Zhong Yang. *Body Sensor Networks*. Computer Science. Springer London, 2014. ISBN 978-1-4471-6373-2. URL <http://www.springer.com/computer/hci/book/978-1-4471-6373-2>.
- [68] Shuo Xiao, A. Dhamdhere, V. Sivaraman, and A. Burdett. Transmission power control in body area sensor networks for healthcare monitoring. *Selected Areas in Communications, IEEE Journal on*, 27(1):37–48, January 2009. ISSN 0733-8716. doi: 10.1109/JSAC.2009.090105.
- [69] Hande Alemdar and Cem Ersoy. Wireless sensor networks for healthcare: A survey. *Computer Networks*, 54(15):2688 – 2710, 2010. ISSN 1389-1286. doi: <http://dx.doi.org/10.1016/j.comnet.2010.05.003>. URL <http://www.sciencedirect.com/science/article/pii/S1389128610001398>.
- [70] Theodore Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition, 2001. ISBN 0130422320.
- [71] Dimitrios Lymberopoulos, Quentin Lindsey, and Andreas Savvides. An empirical characterization of radio signal strength variability in 3-d ieee 802.15.4 networks using monopole antennas. In Kay Römer, Holger Karl, and Friedemann Mattern, editors, *Wireless Sensor Networks*, volume 3868 of *Lecture Notes in Computer Science*, pages 326–341. Springer Berlin Heidelberg, 2006. ISBN 978-3-540-32158-3. doi: 10.1007/11669463_24. URL http://dx.doi.org/10.1007/11669463_24.
- [72] Emiliano Miluzzo, Xiao Zheng, Kristóf Fodor, and Andrew T. Campbell. Radio characterization of 802.15.4 and its impact on the design of mobile sensor

- networks. In *In Proc. 5th European Conf. on Wireless Sensor Networks (EWSN '08)*, pp 171–188, 2008.
- [73] Kannan Srinivasan and Philip Levis. Rssi is under appreciated. In *In Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets)*, 2006.
- [74] Victor Shnayder, Bor-rong Chen, Konrad Lorincz, Thaddeus R. F. Fulford Jones, and Matt Welsh. Sensor networks for medical care. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, SenSys '05*, pages 314–314, New York, NY, USA, 2005. ACM. ISBN 1-59593-054-X. doi: 10.1145/1098918.1098979. URL <http://doi.acm.org/10.1145/1098918.1098979>.
- [75] F.A. Tobagi and L. Kleinrock. Packet switching in radio channels: Part ii—the hidden terminal problem in carrier sense multiple-access and the busy-tone solution. *Communications, IEEE Transactions on*, 23(12):1417–1433, Dec 1975. ISSN 0090-6778. doi: 10.1109/TCOM.1975.1092767.
- [76] Sunil Kandukuri and Stephen Boyd. Optimal power control in interference-limited fading wireless channels with outage-probability specifications. *IEEE Transactions on Wireless Communications*, 1(1):46–55, 2002. URL <http://dblp.uni-trier.de/db/journals/twc/twc1.html#KandukuriB02>.
- [77] Approved draft revision for iee standard for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements-part 15.4b: Wireless medium access control (mac) and physical layer (phy) specifications for low rate wireless personal area networks (wpans) (amendment of iee std 802.15.4-2003). *IEEE Std P802.15.4/D6*, pages –, 2006.
- [78] Draft amendment to iee standard for information technology-telecommunications and information exchange between systems-part 15.4:wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (lr-wpans): Amendment to add alternate phy (amendment of iee std 802.15.4). *IEEE Std P802.15.4a/D5*, pages –, 2006.

- [79] Approved draft amendment to IEEE standard for information technology-telecommunications and information exchange between systems-part 15.4:wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANS): Amendment to add alternate PHY (amendment of IEEE Std 802.15.4). *IEEE Approved Std P802.15.4a/D7*, Jan 2007, pages –, 2007.
- [80] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler. Tinyos: An operating system for sensor networks. In Werner Weber, JanM. Rabaey, and Emile Aarts, editors, *Ambient Intelligence*, pages 115–148. Springer Berlin Heidelberg, 2005. ISBN 978-3-540-23867-6. doi: 10.1007/3-540-27139-2_7. URL http://dx.doi.org/10.1007/3-540-27139-2_7.
- [81] B. Zurita Ares, P.G. Park, C. Fischione, A. Speranzon, and K.H. Johansson. On power control for wireless sensor networks: System model, middleware component and experimental evaluation. In *European Control Conference*, 2007.
- [82] Pan Gun Park, Carlo Fischione, and Karl Henrik Johansson. Experimental evaluation of power control algorithms for wireless sensor networks. In *Proceedings of the 17th World Congress The International Federation of Automatic Control Seoul, Korea, July 6-11, 2008*, 2008.
- [83] S.M. Mahdi Alavi, Michael J. Walsh, and Martin J. Hayes. Robust distributed active power control technique for {IEEE} 802.15.4 wireless sensor networks — a quantitative feedback theory approach. *Control Engineering Practice*, 17(7):805 – 814, 2009. ISSN 0967-0661. doi: <http://dx.doi.org/10.1016/j.conengprac.2009.02.001>. URL <http://www.sciencedirect.com/science/article/pii/S0967066109000161>.
- [84] *CC2420 datasheet*, rev. b edition, 2007. URL <http://www.ti.com/lit/ds/symlink/cc2420.pdf>. Datasheet.
- [85] Basil Kouvaritakis, J. Anthony Rossiter, and Jan Schuurmans. Efficient robust predictive control. *IEEE Transactions on Automatic Control*, 45(8):1545–1549, 2000.
- [86] T. Alamo, D. Muñoz de la Peña, D. Limon, and E.F. Camacho. Constrained min-max predictive control: modifications of the objective function leading to

- polynomial complexity. *Automatic Control, IEEE Transactions on*, 50(5):710–714, May 2005. ISSN 0018-9286. doi: 10.1109/TAC.2005.847039.
- [87] H.S. Witsenhausen. A minimax control problem for sampled linear systems. *Automatic Control, IEEE Transactions on*, 13(1):5–21, Feb 1968. ISSN 0018-9286. doi: 10.1109/TAC.1968.1098788.
- [88] D.M. de la Peña, T. Alamo, D.R. Ramirez, and E.F. Camacho. Min-max model predictive control as a quadratic program. *Control Theory Applications, IET*, 1(1):328–333, January 2007. ISSN 1751-8644. doi: 10.1049/iet-cta:20060016.
- [89] Moteiv. *Tmote Sky Datasheet*, 2006. URL <http://www.eecs.harvard.edu/~konrad/projects/shimmer/references/tmote-sky-datasheet.pdf>.
- [90] D. Muñoz de la Peña, D.R. Ramírez, E.F. Camacho, and T. Alamo. Application of an explicit min-max {MPC} to a scaled laboratory process. *Control Engineering Practice*, 13(12):1463 – 1471, 2005. ISSN 0967-0661. doi: <http://dx.doi.org/10.1016/j.conengprac.2004.12.008>. URL <http://www.sciencedirect.com/science/article/pii/S0967066105000158>. Special Section on Power Plants and Power Systems Control Special Section on Power Plants and Power Systems Control.
- [91] D. Muñoz de la Peña, D.R. Ramírez, E.F. Camacho, and T. Alamo. Explicit solution of min–max {MPC} with additive uncertainties and quadratic criterion. *Systems & Control Letters*, 55(4):266 – 274, 2006. ISSN 0167-6911. doi: <http://dx.doi.org/10.1016/j.sysconle.2005.08.006>. URL <http://www.sciencedirect.com/science/article/pii/S0167691105001325>.
- [92] H.J. Su and E. Geraniotis. Adaptive closed-loop power control with quantized feedback and loop filtering. *Wireless Communications, IEEE Transactions on*, 1(1):76–86, Jan 2002. ISSN 1536-1276. doi: 10.1109/7693.975447.
- [93] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338 – 353, 1965. ISSN 0019-9958. doi: [http://dx.doi.org/10.1016/S0019-9958\(65\)90241-X](http://dx.doi.org/10.1016/S0019-9958(65)90241-X). URL <http://www.sciencedirect.com/science/article/pii/S001999586590241X>.
- [94] E.H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1 – 13, 1975. ISSN 0020-7373. doi: [http://dx.doi.org/10.1016/S0020-7373\(75\)](http://dx.doi.org/10.1016/S0020-7373(75))

- 80002-2. URL <http://www.sciencedirect.com/science/article/pii/S0020737375800022>.
- [95] Werner Van Leekwijck and Etienne E. Kerre. Defuzzification: criteria and classification. *Fuzzy Sets and Systems*, 108(2):159 – 178, 1999. ISSN 0165-0114. doi: [http://dx.doi.org/10.1016/S0165-0114\(97\)00337-0](http://dx.doi.org/10.1016/S0165-0114(97)00337-0). URL <http://www.sciencedirect.com/science/article/pii/S0165011497003370>.
- [96] Tomohiro Takagi and Michio Sugeno. Fuzzy Identification of Systems and Its Applications to Modeling and Control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15(1):116–132, February 1985. URL <http://www.hi.cs.meiji.ac.jp/~takagi/paper/TS-MODEL.tar.gz>.
- [97] K. Kiriakidis. Fuzzy model-based control of complex plants. *Fuzzy Systems, IEEE Transactions on*, 6(4):517 –529, nov 1998. ISSN 1063-6706. doi: 10.1109/91.728444.
- [98] J.S.R. Jang. Anfis: adaptive-network-based fuzzy inference system. *Systems, Man and Cybernetics, IEEE Transactions on*, 23(3):665–685, May 1993. ISSN 0018-9472. doi: 10.1109/21.256541.
- [99] J.S.R. Jang, C.T. Sun, and E. Mizutani. *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*. MATLAB curriculum series. ISBN 9780132610667.
- [100] International Organization for Standardization (ISO). *Automation systems and integration -Integration of advanced process control and optimization capabilities for manufacturing systems -Part 1: Framework and functional model*. ISO, 2013.
- [101] Schneider Electric. Fuzzy control library v1.2. Technical Report 33004219.02, Schneider Electric SA, 35 rue Joseph Monier 92500 Rueil Malmaison - France, February 2009. URL http://download.schneider-electric.com/files?p_File_Id=27511099&p_File_Name=33004219_K01_000_02.pdf.
- [102] J.M. Escaño, J.C. Stockdale, J. Liang, and A. Vakaloudis. Fuzzy matching engine for non-textual authentication: A case study. In *Science and Information Conference (SAI), 2014*, pages 615–619, Aug 2014. doi: 10.1109/SAI.2014.6918250.
- [103] J. Stockdale, A. Vakaloudis, J.M. Escaño, J. Liang, and B. Cahill. A fuzzy system for three-factor, non-textual authentication. In Kohei Arai, Supriya

- Kapoor, and Rahul Bhatia, editors, *Intelligent Systems in Science and Information 2014*, volume 591 of *Studies in Computational Intelligence*, pages 125–137. Springer International Publishing, 2015. ISBN 978-3-319-14653-9. doi: 10.1007/978-3-319-14654-6_8. URL http://dx.doi.org/10.1007/978-3-319-14654-6_8.
- [104] Z. Chi, H. Yan, and T. Pham. *Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition*. Advances in fuzzy systems - applications and theory. ISBN 9789810226978.
- [105] Ferdinando Di Martino and Salvatore Sessa. Image matching by using fuzzy transforms. 2013. doi: 10.1155/2013/760704.
- [106] Irina Perfilieva. Fuzzy transforms. In JamesF. Peters, Andrzej Skowron, Didier Dubois, JerzyW. Grzymala-Busse, Masahiro Inuiguchi, and Lech Polkowski, editors, *Transactions on Rough Sets II*, volume 3135 of *Lecture Notes in Computer Science*, pages 63–81. Springer Berlin Heidelberg, 2005. ISBN 978-3-540-23990-1. doi: 10.1007/978-3-540-27778-1_5. URL http://dx.doi.org/10.1007/978-3-540-27778-1_5.
- [107] I. Bloch. Fuzzy relative position between objects in image processing: a morphological approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(7):657–664, Jul 1999. ISSN 0162-8828. doi: 10.1109/34.777378.
- [108] Isabelle Bloch and Anca Ralescu. Directional relative position between objects in image processing: a comparison between fuzzy approaches. *Pattern Recognition*, 36(7):1563 – 1582, 2003. ISSN 0031-3203. doi: [http://dx.doi.org/10.1016/S0031-3203\(02\)00263-7](http://dx.doi.org/10.1016/S0031-3203(02)00263-7). URL <http://www.sciencedirect.com/science/article/pii/S0031320302002637>.
- [109] Q. Tan and M. Akimoto. Fuzzy matching for robot localization. In *Intelligent Robots and Systems '96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, volume 3, pages 1165–1170 vol.3, Nov 1996. doi: 10.1109/IROS.1996.568966.
- [110] Haiyuan Wu, Qian Chen, and M. Yachida. Face detection from color images using a fuzzy pattern matching method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(6):557–563, Jun 1999. ISSN 0162-8828. doi: 10.1109/34.771326.

- [111] Tian Jinwen, Huang Jianzhong, Liu Jian, and Li Dchua. Image matching based on fuzzy information. In *Signal Processing, 1996., 3rd International Conference on*, volume 2, pages 946–949 vol.2, Oct 1996. doi: 10.1109/ICSIGP.1996.566246.
- [112] Esko Ukkonen. Algorithms for approximate string matching. *Information and Control*, 64(1–3):100 – 118, 1985. ISSN 0019-9958. doi: [http://dx.doi.org/10.1016/S0019-9958\(85\)80046-2](http://dx.doi.org/10.1016/S0019-9958(85)80046-2). URL <http://www.sciencedirect.com/science/article/pii/S0019995885800462>. International Conference on Foundations of Computation Theory.
- [113] Alberto Apostolico and Zvi Galil, editors. *Pattern matching algorithms*. Oxford University Press, New York, 1997. ISBN 0-19-511367-5. URL <http://opac.inria.fr/record=b1092702>.
- [114] Surajit Chaudhuri, Kris Ganjam, Venkatesh Ganti, and Rajeev Motwani. Robust and efficient fuzzy match for online data cleaning. In *In SIGMOD*, pages 313–324, 2003.
- [115] Adnan A.Y. Mustafa. Fuzzy shape matching with boundary signatures. *Pattern Recognition Letters*, 23(12):1473 – 1482, 2002. ISSN 0167-8655. doi: [http://dx.doi.org/10.1016/S0167-8655\(02\)00107-1](http://dx.doi.org/10.1016/S0167-8655(02)00107-1). URL <http://www.sciencedirect.com/science/article/pii/S0167865502001071>.
- [116] Zhao kang Li, Liang jie Xu, Juan Fang, Qun jie Peng, and Min Wang. Research on the surrounding traffic flow of railway station based on license plate recognition and fuzzy matching. In *Business Management and Electronic Information (BMEI), 2011 International Conference on*, volume 4, pages 873–876, May 2011. doi: 10.1109/ICBMEI.2011.5914243.
- [117] P.T. Schultz and R.A. Sartini. Method and system for multi-factor biometric authentication.
- [118] Ritesh Vyas and Gourav Garg. Face recognition using feature extraction and neuro-fuzzy techniques. *International Journal of Electronics and Computer Science Engineering*, 1(4):2048–2056, 2012. ISSN 2277-1956.
- [119] Itseez. Opencv, open source computer vision, 2015. URL <http://opencv.org/>.

- [120] B. Bhanu and V. Govindaraju. *Multibiometrics for Human Identification*. ISBN 9781139501804.
- [121] Asli Celikyilmaz and I. Burhan Trksen. *Modeling Uncertainty with Fuzzy Logic: With Recent Theory and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2009. ISBN 3540899235, 9783540899235.
- [122] Andras Bardossy and Lucien Duckstein. *Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological, and Engineering Systems*. CRC Press, Inc., Boca Raton, FL, USA, 1995. ISBN 0849378338.
- [123] Oscar Cordón, Francisco Herrera, Frank Hoffmann, Luis Magdalena, O Cordon, F Herrera, and F Hoffmann. *Genetic fuzzy systems*. World Scientific Publishing Company Singapore, 2001.
- [124] Niel Canty. *A new algorithm for piecewise affine system identification and its application to a MEMS switch*. PhD thesis, Cork Institute of Technology, Rossa Avenue, Bishopstown. Cork, 2014.
- [125] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943. ISSN 0007-4985. doi: 10.1007/BF02478259. URL <http://dx.doi.org/10.1007/BF02478259>.
- [126] Chao-Chee Ku and Kwang Y Lee. Diagonal recurrent neural networks for dynamic systems control. *Neural Networks, IEEE Transactions on*, 6(1):144–156, 1995.
- [127] Ching-Hung Lee and Ching-Cheng Teng. Identification and control of dynamic systems using recurrent fuzzy neural networks. *Fuzzy Systems, IEEE Transactions on*, 8(4):349–366, 2000.
- [128] Arshdeep Kaur and Amrit Kaur. Comparison of mamdani-type and sugeno-type fuzzy inference systems for air conditioning system. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(2), May 2012. ISSN 2231-2307.
- [129] S. Chiu. Fuzzy Model Identification based on cluster estimation. *Journal of Intelligent Fuzzy Systems*, 2:267–278, 1994.
- [130] Stephen L Chiu. Selecting input variables for fuzzy models. *Journal of Intelligent and Fuzzy Systems*, 4(4):243–256, 1996.

- [131] R. Sindelar and R. Babuska. Input selection for nonlinear regression models. *Fuzzy Systems, IEEE Transactions on*, 12(5):688–696, Oct 2004. ISSN 1063-6706. doi: 10.1109/TFUZZ.2004.834810.
- [132] J. Yen, Haojin Wang, and J. Liao. A method for automatic generation of a fuzzy model. In *Industrial Fuzzy Control and Intelligent Systems, 1993., IFIS '93., Third International Conference on*, pages 88–92, Dec 1993. doi: 10.1109/IFIS.1993.324208.
- [133] T.W. Anderson. *An Introduction to Multivariate Statistical Analysis*. Wiley Series in Probability and Statistics. ISBN 9780471360919.
- [134] Michael C. Mackey, Leon Glass, et al. Oscillation and chaos in physiological control systems. *Science*, 197(4300):287–289, 1977.
- [135] A. S. Weigend and N. A. Gershenfeld, editors. *Time series prediction: Forecasting the future and understanding the past*, 1994. URL http://adsabs.harvard.edu/cgi-bin/nph-bib_query?bibcode=1994tspf.conf.....W.
- [136] Pentti Saikkonen. Introduction to modern time series analysis by gebhard kirchgässner, jürgen wolters. *International Statistical Review*, 76(1):151–152, 2008. ISSN 1751-5823. doi: 10.1111/j.1751-5823.2007.00039_13.x. URL http://dx.doi.org/10.1111/j.1751-5823.2007.00039_13.x.
- [137] Richard McCleary, Richard A. Hay, Errol E. Meidinger, and David McDowall. *Applied time series analysis for the social sciences*. Sage Publications Beverly Hills, CA, 1980.
- [138] Scott L. Zeger, Rafael Irizarry, and Roger D. Peng. On time series analysis of public health and biomedical data. *Annu. Rev. Public Health*, 27:57–79, 2006.
- [139] Qiang Song and Brad S. Chissom. Forecasting enrollments with fuzzy time series—part i. *Fuzzy Sets Syst.*, 54(1):1–9, February 1993. ISSN 0165-0114. doi: 10.1016/0165-0114(93)90355-L. URL [http://dx.doi.org/10.1016/0165-0114\(93\)90355-L](http://dx.doi.org/10.1016/0165-0114(93)90355-L).
- [140] Qiang Song and Brad S. Chissom. Fuzzy time series and its models. *Fuzzy Sets Syst.*, 54(3):269–277, March 1993. ISSN 0165-0114. doi: 10.1016/0165-0114(93)90372-O. URL [http://dx.doi.org/10.1016/0165-0114\(93\)90372-O](http://dx.doi.org/10.1016/0165-0114(93)90372-O).

- [141] Qiang Song and Brad S. Chissom. Forecasting enrollments with fuzzy time series—part ii. *Fuzzy Sets Syst.*, 62(1):1–8, February 1994. ISSN 0165-0114. doi: 10.1016/0165-0114(94)90067-1. URL [http://dx.doi.org/10.1016/0165-0114\(94\)90067-1](http://dx.doi.org/10.1016/0165-0114(94)90067-1).
- [142] F.J.J. Dos Santos and H. De Arruda Camargo. Preprocessing in fuzzy time series to improve the forecasting accuracy. In *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, volume 2, pages 170–173, Dec 2013. doi: 10.1109/ICMLA.2013.185.
- [143] Shyi-Ming Chen. Forecasting enrollments based on fuzzy time series. *Fuzzy Sets Syst.*, 81(3):311–319, August 1996. ISSN 0165-0114. doi: 10.1016/0165-0114(95)00220-0. URL [http://dx.doi.org/10.1016/0165-0114\(95\)00220-0](http://dx.doi.org/10.1016/0165-0114(95)00220-0).
- [144] Shyi-Ming Chen and Nien-Yi Chung. Forecasting enrollments using high-order fuzzy time series and genetic algorithms: Research articles. *Int. J. Intell. Syst.*, 21(5):485–501, May 2006. ISSN 0884-8173. doi: 10.1002/int.v21:5. URL <http://dx.doi.org/10.1002/int.v21:5>.
- [145] Shyi-Ming Chen and Chia-Ching Hsu. A new method to forecast enrollments using fuzzy time series. *International Journal of Applied Science and Engineering*, 2(3):234–244, 2004.
- [146] Steffen F. Bocklisch and Michael Päßler. Fuzzy time series analysis. In Rainer Hampel, Michael Wagenknecht, and Nasredin Chaker, editors, *Fuzzy Control*, volume 6 of *Advances in Soft Computing*, pages 331–345. Physica-Verlag HD, 2000. ISBN 978-3-7908-1327-2. doi: 10.1007/978-3-7908-1841-3_30. URL http://dx.doi.org/10.1007/978-3-7908-1841-3_30.
- [147] P. J. Werbos. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD thesis, Harvard University, 1974.
- [148] A. Ghosh and M. Chakraborty. Hybrid optimized back propagation learning algorithm for multi-layer perceptron. *International Journal of Computer Applications*, 57, 2012.
- [149] Brian S. Everitt, Sabine Landau, and Morven Leese. *Cluster Analysis*. Wiley Publishing, 4th edition, 2009. ISBN 0340761199, 9780340761199.

- [150] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Comput. Surv.*, 31(3):264–323, September 1999. ISSN 0360-0300. doi: 10.1145/331499.331504. URL <http://doi.acm.org/10.1145/331499.331504>.
- [151] Pavel Berkhin. Survey of clustering data mining techniques. Technical report, 2002.
- [152] Rui Xu and II Wunsch, D. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645–678, May 2005. ISSN 1045-9227. doi: 10.1109/TNN.2005.845141.
- [153] Igor V. Tetko, David J. Livingstone, and Alexander I. Luik. Neural network studies. 1. comparison of overfitting and overtraining. *Journal of Chemical Information and Computer Sciences*, 35(5):826–833, 1995. doi: 10.1021/ci00027a006. URL <http://dx.doi.org/10.1021/ci00027a006>.
- [154] S. Kollias and D. Anastassiou. Adaptive training of multilayer neural networks using a least squares estimation technique. In *Neural Networks, 1988., IEEE International Conference on*, pages 383–390 vol.1, July 1988. doi: 10.1109/ICNN.1988.23870.
- [155] L.P. Ricotti, S. Ragazzini, and G. Martinelli. Learning of word stress in a sub-optimal second order back-propagation neural network. In *Neural Networks, 1988., IEEE International Conference on*, pages 355–361 vol.1, July 1988. doi: 10.1109/ICNN.1988.23867.
- [156] Michael C. Mozer and Paul Smolensky. Using relevance to reduce network size automatically. *Connection Science*, 1(1):3–16, 1989. doi: 10.1080/09540098908915626. URL <http://dx.doi.org/10.1080/09540098908915626>.
- [157] M.G. Bello. Enhanced training algorithms, and integrated training/architecture selection for multilayer perceptron networks. *Neural Networks, IEEE Transactions on*, 3(6):864–875, Nov 1992. ISSN 1045-9227. doi: 10.1109/72.165589.
- [158] E.D. Karnin. A simple procedure for pruning back-propagation trained neural networks. *Neural Networks, IEEE Transactions on*, 1(2):239–242, Jun 1990. ISSN 1045-9227. doi: 10.1109/72.80236.

- [159] Guozhong Zhou and Sun Yaming. A combined gradient learning algorithm for multilayered neural networks. In *Industrial Electronics, Control and Instrumentation, 1991. Proceedings. IECON '91., 1991 International Conference on*, pages 1492–1495 vol.2, Oct 1991. doi: 10.1109/IECON.1991.239120.
- [160] John H. Holland. *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press, 1975.
- [161] Zbigniew Michalewicz. *Genetic algorithms+ data structures= evolution programs*. springer, 1996.
- [162] Thomas Bäck, Frank Hoffmeister, and Hans-Paul Schwefel. A survey of evolution strategies. In *Proceedings of the 4th international conference on genetic algorithms*, pages 2–9, 1991.
- [163] David E. Goldberg. Real-coded genetic algorithms, virtual alphabets, and blocking. *Urbana*, 51:61801, 1990.
- [164] Francisco Herrera, Manuel Lozano, and Jose L. Verdegay. Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis. *Artificial intelligence review*, 12(4):265–319, 1998.
- [165] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992. ISBN 0-262-11170-5.
- [166] John R. Koza. *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, Cambridge, MA, USA, 1994. ISBN 0-262-11189-6.
- [167] Antonio A. Alonso, Julio R. Banga, and Ricardo Perez-Martin. A complete dynamic model for the thermal processing of bioproducts in batch units and its application to controller design. *Chemical Engineering Science*, 52(8):1307 – 1322, 1997. ISSN 0009-2509. doi: [http://dx.doi.org/10.1016/S0009-2509\(96\)00484-8](http://dx.doi.org/10.1016/S0009-2509(96)00484-8). URL <http://www.sciencedirect.com/science/article/pii/S0009250996004848>.
- [168] Carlos Vilas, Míriam R. García, Julio R. Banga, and Antonio A. Alonso. Desarrollo de una librería de componentes en ecosimpro para la operación de plantas de procesamiento térmico de alimentos. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 5(1):51–65, 2008.

- [169] J.M. Escaño and C. Bordons. Modelo neuroborroso de la presión de la cámara de mezcla en una fundición de cobre. In *XXVIII Jornadas Técnicas de Automática. CEA-IFAC. 2006*, 2006.
- [170] J.M. Escaño, F. Dorado, and C. Bordons. Pca based pressure control of a gas mixing chamber. In *Emerging Technologies Factory Automation, 2009. ETFA 2009. IEEE Conference on*, pages 1–6, Sept 2009. doi: 10.1109/ETFA.2009.5347023.
- [171] Chi-Huang Lu and Ching-Chih Tsai. Generalized predictive control using recurrent fuzzy neural networks for industrial processes. *Journal of Process Control*, 17(1):83 – 92, 2007. ISSN 0959-1524. doi: <http://dx.doi.org/10.1016/j.jprocont.2006.08.003>. URL <http://www.sciencedirect.com/science/article/pii/S0959152406000904>.
- [172] Stephen L. Chiu. Fuzzy model identification based on cluster estimation. *Journal of intelligent and Fuzzy systems*, 2(3):267–278, 1994.
- [173] J.S.R. Jang and C.T. Sun. Neuro-fuzzy modeling and control. *Proceedings of the IEEE*, 83(3):378–406, 1995.
- [174] K. Kiriakidis. Fuzzy model-based control of complex plants. *Fuzzy Systems, IEEE Transactions on*, 6(4):517–529, Nov 1998. ISSN 1063-6706. doi: 10.1109/91.728444.
- [175] Radu-Emil Precup and Hans Hellendoorn. A survey on industrial applications of fuzzy control. *Computers in Industry*, 62(3):213 – 226, 2011. ISSN 0166-3615. doi: <http://dx.doi.org/10.1016/j.compind.2010.10.001>. URL <http://www.sciencedirect.com/science/article/pii/S0166361510001363>.
- [176] Jan Jantzen. A robustness study of fuzzy control rules. *Technical University of Denmark, Department of Automation, Dinamarca*, 1997.
- [177] Jan Jantzen. Tuning of fuzzy pid controllers. *Technical University of Denmark, Department of Automation, Bldg, 326*, 1998.
- [178] Jan Jantzen. Design of fuzzy controllers. *Technical University of Denmark, Department of Automation, Bldg, 326:362–367*, 1998.
- [179] Jan Jantzen. *Foundations of Fuzzy Control: A Practical Approach*. John Wiley & Sons, 2013.

- [180] J.M. Escaño, M.G. Ortega, and F.R. Rubio. Position control of a pneumatic levitation system. In *Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on*, volume 1, pages 6 pp.–528, Sept 2005. doi: 10.1109/ETFA.2005.1612568.
- [181] D. Algarín. Reconocimiento de objetos mediante visión artificial (in spanish). Technical report, Centro Tecnológico ALTAIR, 2002.
- [182] H. K. Lam, F. H. Frank Leung, and Peter Kwong-Shun Tam. Stable and robust fuzzy control for uncertain nonlinear systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 30(6):825–840, 2000. doi: 10.1109/3468.895910. URL <http://dx.doi.org/10.1109/3468.895910>.
- [183] H. K. Lam, F. H. Frank Leung, and Peter Kwong-Shun Tam. Stability analysis & design of fuzzy observer-controller for fuzzy systems. In *Proceedings of the 10th IEEE International Conference on Fuzzy Systems, Melbourne, Australia, December 2-5, 2001*, pages 1259–1262, 2001. doi: 10.1109/FUZZ.2001.1008887. URL <http://dx.doi.org/10.1109/FUZZ.2001.1008887>.
- [184] H. K. Lam, F. H. Frank Leung, and Peter Kwong-Shun Tam. Linear controllers for fuzzy systems subject to unknowns parameters: Stability analysis and design based on linear matrix inequality (LMI) approach. In *Proceedings of the 10th IEEE International Conference on Fuzzy Systems, Melbourne, Australia, December 2-5, 2001*, pages 1012–1015, 2001. doi: 10.1109/FUZZ.2001.1009133. URL <http://dx.doi.org/10.1109/FUZZ.2001.1009133>.
- [185] H. K. Lam, Sai-Ho Ling, F. H. Frank Leung, and Peter Kwong-Shun Tam. Optimal and stable fuzzy controllers for nonlinear systems subject to parameter uncertainties using genetic algorithm. In *Proceedings of the 10th IEEE International Conference on Fuzzy Systems, Melbourne, Australia, December 2-5, 2001*, pages 908–911, 2001. doi: 10.1109/FUZZ.2001.1009103. URL <http://dx.doi.org/10.1109/FUZZ.2001.1009103>.
- [186] Hak-Keung Lam, Frank Hung-Fat Leung, and Peter Kwong-Shun Tam. Nonlinear state feedback controller for nonlinear systems: Stability analysis and design based on fuzzy plant model. *IEEE T. Fuzzy Systems*, 9(4):657–661, 2001. doi: 10.1109/91.940975. URL <http://dx.doi.org/10.1109/91.940975>.
- [187] Frank Hung-Fat Leung, H. K. Lam, Peter Kwong-Shun Tam, and Yim-Shu Lee. Stable fuzzy controller design for uncertain nonlinear systems: genetic

- algorithm approach. In *The 12th IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2003, St. Louis, Missouri, USA, 25-28 May 2003*, pages 500–505, 2003. doi: 10.1109/FUZZ.2003.1209414. URL <http://dx.doi.org/10.1109/FUZZ.2003.1209414>.
- [188] H. K. Lam and F. H. Frank Leung. Stability analysis of fuzzy control systems subject to uncertain grades of membership. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 35(6):1322–1325, 2005. doi: 10.1109/TSMCB.2005.850181. URL <http://dx.doi.org/10.1109/TSMCB.2005.850181>.
- [189] H. K. Lam and F. H. Frank Leung. Fuzzy controller with stability and performance rules for nonlinear systems. *Fuzzy Sets and Systems*, 158(2):147–163, 2007. doi: 10.1016/j.fss.2006.06.005. URL <http://dx.doi.org/10.1016/j.fss.2006.06.005>.
- [190] Mohammad Narimani and Hak-Keung Lam. Sos-based stability analysis of polynomial fuzzy control systems via polynomial membership functions. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, USA, 11-14 October 2009*, pages 3011–3016, 2009. doi: 10.1109/ICSMC.2009.5345897. URL <http://dx.doi.org/10.1109/ICSMC.2009.5345897>.
- [191] H. K. Lam, Mohammad Narimani, and Lakmal D. Seneviratne. Quadratic stability analysis of fuzzy control systems using stepwise membership functions. In *FUZZ-IEEE 2009, IEEE International Conference on Fuzzy Systems, Jeju Island, Korea, 20-24 August 2009, Proceedings*, pages 790–795, 2009. doi: 10.1109/FUZZY.2009.5277358. URL <http://dx.doi.org/10.1109/FUZZY.2009.5277358>.
- [192] Mohammad Narimani and Hak-Keung Lam. Relaxed lmi-based stability conditions for takagi-sugeno fuzzy control systems using regional-membership-function-shape-dependent analysis approach. *IEEE T. Fuzzy Systems*, 17(5): 1221–1228, 2009. doi: 10.1109/TFUZZ.2009.2025959. URL <http://dx.doi.org/10.1109/TFUZZ.2009.2025959>.
- [193] H. K. Lam and Shun-Hung Tsai. Stability analysis of polynomial-fuzzy-model-based control systems with mismatched premise membership functions. *IEEE T. Fuzzy Systems*, 22(1):223–229, 2014. doi: 10.1109/TFUZZ.2013.2243735. URL <http://dx.doi.org/10.1109/TFUZZ.2013.2243735>.

- [194] H. K. Lam, Mohammad Narimani, Hongyi Li, and Honghai Liu. Stability analysis of polynomial-fuzzy-model-based control systems using switching polynomial lyapunov function. *IEEE T. Fuzzy Systems*, 21(5):800–813, 2013. doi: 10.1109/TFUZZ.2012.2230005. URL <http://dx.doi.org/10.1109/TFUZZ.2012.2230005>.
- [195] Hak-Keung Lam. Lmi-based stability analysis for fuzzy-model-based control systems using artificial T-S fuzzy model. *IEEE T. Fuzzy Systems*, 19(3):505–513, 2011. doi: 10.1109/TFUZZ.2011.2116027. URL <http://dx.doi.org/10.1109/TFUZZ.2011.2116027>.
- [196] H. K. Lam, F. H. Frank Leung, and Peter Kwong-Shun Tam. Design and stability analysis of fuzzy model-based nonlinear controller for nonlinear systems using genetic algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 33(2):250–257, 2003. doi: 10.1109/TSMCB.2003.810440. URL <http://dx.doi.org/10.1109/TSMCB.2003.810440>.
- [197] H. K. Lam and Frank Hung-Fat Leung. LMI relaxed stability conditions for fuzzy-model-based control systems. In *IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2006, Vancouver, BC, Canada, July 16-21, 2006*, pages 1932–1938, 2006. doi: 10.1109/FUZZY.2006.1681968. URL <http://dx.doi.org/10.1109/FUZZY.2006.1681968>.
- [198] Hak-Keung Lam, Chun Wan Yeung, and F. H. Frank Leung. Stability analysis and performance design for fuzzy-model-based control system under imperfect premise matching. In *FUZZ-IEEE 2007, IEEE International Conference on Fuzzy Systems, Imperial College, London, UK, 23-26 July, 2007, Proceedings*, pages 1–6, 2007. doi: 10.1109/FUZZY.2007.4295423. URL <http://dx.doi.org/10.1109/FUZZY.2007.4295423>.
- [199] H. K. Lam and Lakmal D. Seneviratne. Bmi-based stability and performance design for fuzzy-model-based control systems subject to parameter uncertainties. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 37(3): 502–514, 2007. doi: 10.1109/TSMCB.2006.885308. URL <http://doi.ieeecomputersociety.org/10.1109/TSMCB.2006.885308>.
- [200] H. K. Lam, Mohammad Narimani, Johnny C. Y. Lai, and F. H. Frank Leung. Stability analysis of T-S fuzzy-model-based control systems using fuzzy lyapunov function. In *FUZZ-IEEE 2008, IEEE International Conference on*

- Fuzzy Systems, Hong Kong, China, 1-6 June, 2008, Proceedings*, pages 931–938, 2008. doi: 10.1109/FUZZY.2008.4630481. URL <http://dx.doi.org/10.1109/FUZZY.2008.4630481>.
- [201] Hak-Keung Lam and F. H. Frank Leung. Stability analysis of discrete-time fuzzy-model-based control systems with time delay: Time delay-independent approach. *Fuzzy Sets and Systems*, 159(8):990–1000, 2008. doi: 10.1016/j.fss.2007.11.012. URL <http://dx.doi.org/10.1016/j.fss.2007.11.012>.
- [202] Hak-Keung Lam and Mohammad Narimani. Sum-of-squares-based stability analysis of polynomial fuzzy-model-based control systems. In *FUZZ-IEEE 2009, IEEE International Conference on Fuzzy Systems, Jeju Island, Korea, 20-24 August 2009, Proceedings*, pages 234–239, 2009. doi: 10.1109/FUZZY.2009.5277416. URL <http://dx.doi.org/10.1109/FUZZY.2009.5277416>.
- [203] Ginny Y. Wong, Frank H. Leung, and Hak-Keung Lam. Relaxed stability conditions for discrete-time fuzzy-model-based control systems. In *FUZZ-IEEE 2009, IEEE International Conference on Fuzzy Systems, Jeju Island, Korea, 20-24 August 2009, Proceedings*, pages 221–226, 2009. doi: 10.1109/FUZZY.2009.5277073. URL <http://dx.doi.org/10.1109/FUZZY.2009.5277073>.
- [204] Hak-Keung Lam and Frank Hung-Fat Leung. *Stability Analysis of Fuzzy-Model-Based Control Systems - Linear-Matrix-Inequality Approach*, volume 264 of *Studies in Fuzziness and Soft Computing*. Springer, 2010. ISBN 978-3-642-17843-6. doi: 10.1007/978-3-642-17844-3. URL <http://dx.doi.org/10.1007/978-3-642-17844-3>.
- [205] Antonio Javier Barragán Piña. *Síntesis de sistemas de control borroso estables por diseño*. 2009.
- [206] José M. Andújar, Antonio J. Barragán, Juan M. Córdoba, and I. Fernández. Diseño de sistemas de control borroso: modelado de la planta. *RIAI*, 3(1):75–81, 2010.
- [207] L. X. Wang. Stable adaptive fuzzy control of nonlinear systems. *Trans. Fuz Sys.*, 1(2):146–155, May 1993. ISSN 1063-6706. doi: 10.1109/91.227383. URL <http://dx.doi.org/10.1109/91.227383>.
- [208] Li-Xin Wang. *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1994. ISBN 0-13-099631-9.

- [209] M. Sugeno and G.T. Kang. Fuzzy modelling and control of multilayer incinerator. *Fuzzy Sets and Systems*, 18(3):329 – 345, 1986. ISSN 0165-0114. doi: [http://dx.doi.org/10.1016/0165-0114\(86\)90010-2](http://dx.doi.org/10.1016/0165-0114(86)90010-2). URL <http://www.sciencedirect.com/science/article/pii/0165011486900102>. Dedicated to the memory of Richard E. Bellman.
- [210] Kazuo Tanaka and Michio Sugeno. Stability analysis and design of fuzzy control systems. *Fuzzy Sets and Systems*, 45(2):135 – 156, 1992. ISSN 0165-0114. doi: [http://dx.doi.org/10.1016/0165-0114\(92\)90113-I](http://dx.doi.org/10.1016/0165-0114(92)90113-I). URL <http://www.sciencedirect.com/science/article/pii/016501149290113I>.
- [211] H.O. Wang, K. Tanaka, and M. Griffin. Parallel distributed compensation of nonlinear systems by takagi-sugeno fuzzy model. In *Fuzzy Systems, 1995. International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium., Proceedings of 1995 IEEE Int*, volume 2, pages 531–538 vol.2, Mar 1995. doi: 10.1109/FUZZY.1995.409737.
- [212] H.O. Wang, K. Tanaka, and M. Griffin. An analytical framework of fuzzy modeling and control of nonlinear systems: stability and design issues. In *American Control Conference, Proceedings of the 1995*, volume 3, pages 2272–2276 vol.3, Jun 1995. doi: 10.1109/ACC.1995.531376.
- [213] Kazuo Tanaka and Hua O. Wang. *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*. John Wiley & Sons, Inc., New York, NY, USA, 2002. ISBN 0471224596.
- [214] K. Tanaka and M. Sugeno. Stability analysis of fuzzy systems using lyapunov's direct method. pages 133–136, 1990.
- [215] K. Tanaka and M. Sano. A robust stabilization problem of fuzzy control systems and its application to backing up control of a truck-trailer. *Fuzzy Systems, IEEE Transactions on*, 2(2):119–134, May 1994. ISSN 1063-6706. doi: 10.1109/91.277961.
- [216] Seiji Yasunobu and Shoji Miyamoto. Automatic train operation system by predictive fuzzy control. *Industrial applications of fuzzy control*, 1(18):1–18, 1985.
- [217] R. E. Bellman and L. A. Zadeh. Decision-making in a fuzzy environment. *Management Science*, 17(4):pp. B141–B164, 1970. ISSN 00251909. URL <http://www.jstor.org/stable/2629367>.

- [218] Joao Miguel da Costa Sousa. Fuzzy model-based control of complex processes. *The Annals of the Marie Curie Fellowship Association*, 1(1):2000, 1999.
- [219] J.M. da Costa Sousa and U. Kaymak. Model predictive control using fuzzy decision functions. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 31(1):54–65, Feb 2001. ISSN 1083-4419. doi: 10.1109/3477.907564.
- [220] K. Belarbi and F. Megri. A stable model-based fuzzy predictive control based on fuzzy dynamic programming. *Fuzzy Systems, IEEE Transactions on*, 15(4): 746–754, Aug 2007. ISSN 1063-6706. doi: 10.1109/TFUZZ.2006.890656.
- [221] J.A. Roubos, S. Mollov, R. Babuška, and H.B. Verbruggen. Fuzzy model-based predictive control using takagi–sugeno models. *International Journal of Approximate Reasoning*, 22(1–2):3 – 30, 1999. ISSN 0888-613X. doi: [http://dx.doi.org/10.1016/S0888-613X\(99\)00020-1](http://dx.doi.org/10.1016/S0888-613X(99)00020-1). URL <http://www.sciencedirect.com/science/article/pii/S0888613X99000201>.
- [222] Ning Li, Shao-Yuan Li, and Yu-Geng Xi. Multi-model predictive control based on the takagi–sugeno fuzzy models: a case study. *Information Sciences*, 165 (3–4):247 – 263, 2004. ISSN 0020-0255. doi: <http://dx.doi.org/10.1016/j.ins.2003.10.011>. URL <http://www.sciencedirect.com/science/article/pii/S0020025503004031>.
- [223] Mai Van Sy and Phan Xuan Minh. Fuzzy model predictive control using takagi-sugeno model. In *Control, Automation and Systems, 2008. ICCAS 2008. International Conference on*, pages 632–637, Oct 2008. doi: 10.1109/ICCAS.2008.4694579.
- [224] L. Dalhoumi, M. Djemel, and M. Chtourou. Fuzzy predictive control based on takagi-sugeno model for nonlinear systems. In *Systems Signals and Devices (SSD), 2010 7th International Multi-Conference on*, pages 1–7, June 2010. doi: 10.1109/SSD.2010.5585552.
- [225] H.N. Nounou and K.M. Passino. Fuzzy model predictive control: techniques, stability issues, and examples. In *Intelligent Control/Intelligent Systems and Semiotics, 1999. Proceedings of the 1999 IEEE International Symposium on*, pages 423–428, 1999. doi: 10.1109/ISIC.1999.796692.
- [226] Gorazd Karer, Gašper Mušič, Igor Škrjanc, and Borut Zupančič. Hybrid fuzzy model-based predictive control of temperature in a batch reactor. *Computers & Chemical Engineering*, 31(12):1552 – 1564, 2007. ISSN 0098-1354.

- doi: <http://dx.doi.org/10.1016/j.compchemeng.2007.01.003>. URL <http://www.sciencedirect.com/science/article/pii/S0098135407000051>.
- [227] João Chaínho, Pedro Pereira, Silviano Rafael, A. J. Pires, Labsei Escola, and Superior Tecnologia Setúbal. A simple pid controller with adaptive parameter in a dspic: A case of study. In *In Proceedings of the 9th Spanish-Portuguese Congress on Electrical Engineering*, 2005.
- [228] Xiu Jie and Xia Changliang. Fuzzy logic based adaptive pid control of switched reluctance motor drive. In *Control Conference, 2007. CCC 2007. Chinese*, pages 41–45, July 2007. doi: 10.1109/CHICC.2006.4347328.
- [229] M. Rabbani, H.M.M. Maruf, T. Ahmed, M.A. Kabir, and U. Mahbub. Fuzzy logic driven adaptive pid controller for pwm based buck converter. In *Informatics, Electronics Vision (ICIEV), 2012 International Conference on*, pages 958–962, May 2012. doi: 10.1109/ICIEV.2012.6317492.
- [230] Dnyaneshwar S. Karanjkar, S. Chatterji, Amod Kumar, and S.L. Shimi. Fuzzy adaptive proportional-integral-derivative controller with dynamic set-point adjustment for maximum power point tracking in solar photovoltaic system. *Systems Science and amp; Control Engineering*, 2(1):562–582, 2014. doi: 10.1080/21642583.2014.956267. URL <http://dx.doi.org/10.1080/21642583.2014.956267>.
- [231] Emad Ali. Automatic tuning of model predictive controllers based on fuzzy logic. *optimization*, 2(2):1, 2001.
- [232] Emad Ali and Ashraf Al-Ghazzawi. On-line tuning of model predictive controllers using fuzzy logic. *The Canadian Journal of Chemical Engineering*, 81(5):1041–1051, 2003. ISSN 1939-019X. doi: 10.1002/cjce.5450810515. URL <http://dx.doi.org/10.1002/cjce.5450810515>.
- [233] Emad Ali. Heuristic on-line tuning for nonlinear model predictive controllers using fuzzy logic. *Journal of Process Control*, 13(5):383–396, 2003.
- [234] J. Mendes, R. Araújo, and F. Souza. Adaptive fuzzy generalized predictive control based on discrete-time t-s fuzzy model. In *Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on*, pages 1–8, Sept 2010. doi: 10.1109/ETFA.2010.5641301.

- [235] David R. Vinson. Air separation control technology. *Computers & Chemical Engineering*, 30(10–12):1436 – 1446, 2006. ISSN 0098-1354. doi: <http://dx.doi.org/10.1016/j.compchemeng.2006.05.038>. URL <http://www.sciencedirect.com/science/article/pii/S0098135406001554>. Papers from Chemical Process Control {VII} {CPC} {VII} Seventh international conference in the Series.
- [236] Gang Feng. A survey on analysis and design of model-based fuzzy control systems. *Fuzzy Systems, IEEE Transactions on*, 14(5):676–697, Oct 2006. ISSN 1063-6706. doi: 10.1109/TFUZZ.2006.883415.
- [237] M. Mahfouf, M.F. Abbod, and D.A. Linkens. A survey of fuzzy logic monitoring and control utilisation in medicine. *Artificial Intelligence in Medicine*, 21(1–3): 27 – 42, 2001. ISSN 0933-3657. doi: [http://dx.doi.org/10.1016/S0933-3657\(00\)00072-5](http://dx.doi.org/10.1016/S0933-3657(00)00072-5). URL <http://www.sciencedirect.com/science/article/pii/S0933365700000725>. Fuzzy Theory in Medicine.
- [238] Said Leghmizi and Sheng Liu. A survey of fuzzy control for stabilized platforms. *CoRR*, abs/1109.0428, 2011. URL <http://arxiv.org/abs/1109.0428>.
- [239] S.J. Qin and T.A. Badgwell. An overview of nonlinear model predictive control applications. In Frank Allgöwer and Alex Zheng, editors, *Nonlinear Model Predictive Control*, volume 26 of *Progress in Systems and Control Theory*, pages 369–392. Birkhäuser Basel, 2000. ISBN 978-3-0348-9554-5. doi: 10.1007/978-3-0348-8407-5_21. URL http://dx.doi.org/10.1007/978-3-0348-8407-5_21.
- [240] Thomas A. Badgwell. Review of nonlinear model predictive control applications.
- [241] B. Satpati, C. Koley, P.S. Bhowmik, and S. Datta. Nonlinear model predictive control of pneumatic conveying and drying process. In *Control Applications (CCA), 2014 IEEE Conference on*, pages 492–497, Oct 2014. doi: 10.1109/CCA.2014.6981394.
- [242] B. Potocnik, G. Music, and B. Zupancic. Model predictive control systems with discrete inputs. In *Electrotechnical Conference, 2004. MELECON 2004. Proceedings of the 12th IEEE Mediterranean*, volume 1, pages 383–386 Vol.1, May 2004. doi: 10.1109/MELCON.2004.1346886.

- [243] Alfredo Núñez, Doris Sáez, Simon Oblak, and Igor Škrjanc. Fuzzy-model-based hybrid predictive control. *{ISA} Transactions*, 48(1):24 – 31, 2009. ISSN 0019-0578. doi: <http://dx.doi.org/10.1016/j.isatra.2008.10.007>. URL <http://www.sciencedirect.com/science/article/pii/S0019057808000682>.
- [244] R. Babuska, J. Sousa, and H. Verbruggen. Predictive control of nonlinear systems based on fuzzy and neural models. *European control conference*, page 667, 1999.
- [245] P. Marusak and P. Tatjewski. Stability analysis of nonlinear control systems with unconstrained fuzzy predictive controllers. *Archives of Control Sciences*, Vol. 12, no. 3:267–288, 2002.
- [246] P. Tatjewski. *Advanced control of industrial processes: structures and algorithms*. Advances in industrial control. ISBN 9781846286346.
- [247] Antonio A. Alonso, R.I. Perez-Martin, N.V. Shukla, and P.B. Deshpande. On-line quality control of non-linear batch systems: Application to the thermal processing of canned foods. *Journal of Food Engineering*, 19(3):275 – 289, 1993. ISSN 0260-8774. doi: [http://dx.doi.org/10.1016/0260-8774\(93\)90047-N](http://dx.doi.org/10.1016/0260-8774(93)90047-N). URL <http://www.sciencedirect.com/science/article/pii/026087749390047N>.
- [248] Antonio A Alonso, Julio R Banga, Ricardo Isaac Pérez Martín, et al. New strategy for the control of pressure during the cooling stage of the sterilization process in steam retorts: part i. a preliminary study. 1993.
- [249] Antonio A. Alonso, Julio R. Banga, and Ricardo Perez-Martin. Modeling and adaptive control for batch sterilization. *Computers & Chemical Engineering*, 22(3):445 – 458, 1998. ISSN 0098-1354. doi: [http://dx.doi.org/10.1016/S0098-1354\(97\)00250-0](http://dx.doi.org/10.1016/S0098-1354(97)00250-0). URL <http://www.sciencedirect.com/science/article/pii/S0098135497002500>.
- [250] EA Internacional. Ecosimpro, 2015. URL <http://www.ecosimpro.com>.
- [251] B. Kouvaritakis, M. Cannon, and J. A. Rossiter. Non-linear model based predictive control. *International Journal of Control*, 72(10):919–928, 1999. doi: 10.1080/002071799220650. URL <http://dx.doi.org/10.1080/002071799220650>.
- [252] Shane Townsend and George W Irwin. Nonlinear model based predictive control using multiple local models. *IEE Control Engineering series*, pages 223–244, 2001.

- [253] Tor A. Johansen and Bjarne A. Foss. Identification of non-linear system structure and parameters using regime decomposition. *Automatica*, 31(2):321 – 326, 1995. ISSN 0005-1098. doi: [http://dx.doi.org/10.1016/0005-1098\(94\)00096-2](http://dx.doi.org/10.1016/0005-1098(94)00096-2). URL <http://www.sciencedirect.com/science/article/pii/S0005109894000962>.
- [254] Chi-Huang Lu and Ching-Chih Tsai. Generalized predictive control using recurrent fuzzy neural networks for industrial processes. *Journal of Process Control*, 17(1):83 – 92, 2007. ISSN 0959-1524. doi: 10.1016/j.jprocont.2006.08.003. URL <http://www.sciencedirect.com/science/article/pii/S0959152406000904>.
- [255] Sašo Blažič and Igor Škrjanc. Design and stability analysis of fuzzy model-based predictive control – a case study. *Journal of Intelligent and Robotic Systems*, 49(3):279–292, 2007. ISSN 0921-0296. doi: 10.1007/s10846-007-9147-8. URL <http://dx.doi.org/10.1007/s10846-007-9147-8>.
- [256] M. Mahfouf, D.A. Linkens, and M.F. Abbod. Adaptive fuzzy {TSK} model-based predictive control using a carima model structure. *Chemical Engineering Research and Design*, 78(4):590 – 596, 2000. ISSN 0263-8762. doi: <http://dx.doi.org/10.1205/026387600527527>. URL <http://www.sciencedirect.com/science/article/pii/S0263876200719277>. Process Control.
- [257] J. Zhang and A.J. Morris. Long range predictive control of nonlinear processes based on recurrent neuro-fuzzy network models. *Neural Computing & Applications*, 9(1):50–59, 2000. ISSN 0941-0643. doi: 10.1007/s005210070035. URL <http://dx.doi.org/10.1007/s005210070035>.
- [258] JairoJ. Espinosa and Joos Vandewalle. Predictive control using fuzzy models. In Rajkumar Roy, Takeshi Furuhashi, and PravirK. Chawdhry, editors, *Advances in Soft Computing*, pages 187–200. Springer London, 1999. ISBN 978-1-85233-062-0. doi: 10.1007/978-1-4471-0819-1_14. URL http://dx.doi.org/10.1007/978-1-4471-0819-1_14.
- [259] Jairo J. Espinosa, Mohamed Hadjili, Vincent Wertz, and Joos Vandewalle. Predictive control using fuzzy models – comparative study. In *Proc. European Control Conference*, 1999.

- [260] John Soto, Iván Castillo, and Alain Gauthier. Identificación y control de sistemas no lineales mediante clustering y gpc usando modelos difusos takagi-sugeno. *Revista de Ingeniería*, (19):150–157, 2004.
- [261] E. Zafiriou M. Morari. *Robust process control*. Prentice-Hall, 1989.
- [262] Xiao Wu, Jiong Shen, Yiguo Li, and K.Y. Lee. Stable model predictive control based on ts fuzzy model with application to boiler-turbine coordinated system. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 2981–2987, Dec 2011. doi: 10.1109/CDC.2011.6160553.
- [263] Tiejun Zhang, Gang Feng, and Jianhong Lu. Fuzzy constrained min-max model predictive control based on piecewise lyapunov functions. *Fuzzy Systems, IEEE Transactions on*, 15(4):686–698, Aug 2007. ISSN 1063-6706. doi: 10.1109/TFUZZ.2006.889908.
- [264] N. Wada, K. Saito, and M. Saeki. Model predictive control for linear parameter varying systems using parameter dependent lyapunov function. *Circuits and Systems II: Express Briefs, IEEE Transactions on*, 53(12):1446–1450, Dec 2006. ISSN 1549-7747. doi: 10.1109/TCSII.2006.883832.
- [265] BaoCang Ding, HeXu Sun, and Peng Yang. Further studies on lmi-based relaxed stabilization conditions for nonlinear systems in takagi–sugeno’s form. *Automatica*, 42(3):503 – 508, 2006. ISSN 0005-1098. doi: <http://dx.doi.org/10.1016/j.automatica.2005.11.005>. URL <http://www.sciencedirect.com/science/article/pii/S0005109805004085>.
- [266] J. Gómez Ortega and E.F. Camacho. Mobile robot navigation in a partially structured static environment, using neural predictive control. *Control Engineering Practice*, 4(12):1669 – 1679, 1996. ISSN 0967-0661. doi: [http://dx.doi.org/10.1016/S0967-0661\(96\)00184-0](http://dx.doi.org/10.1016/S0967-0661(96)00184-0). URL <http://www.sciencedirect.com/science/article/pii/S0967066196001840>.
- [267] J. Gomez Ortega, E.F. Camacho, and J. Quero. Neural network local navigation of mobile robots in a moving obstacles environment. In Cs. BÁNYÁSZ, editor, *Intelligent Components and Instruments for Control Applications 1994*, IFAC Postprint Volume, pages 247 – 252. Pergamon, Oxford, 1994. ISBN 978-0-08-042234-3. doi: <http://dx.doi.org/10.1016/B978-0-08-042234-3>.

- 50042-4. URL <http://www.sciencedirect.com/science/article/pii/B9780080422343500424>.
- [268] J. Gomez Ortega and E.F. Camacho. Neural predictive control for mobile robot navigation in a partially structured static environment. In *International Federation of Automatic Control. World Congress (13th: 1996: San Francisco Calif.). Proceedings Vol. Q*, 1997.
- [269] Alexander Gegov. *Complexity Management in Fuzzy Systems - A Rule Base Compression Approach*, volume 211 of *Studies in Fuzziness and Soft Computing*. Springer, 2007. ISBN 978-3-540-38883-8.
- [270] Young-Jeng Chen and Ching-Cheng Teng. Rule combination in a fuzzy neural network. *Fuzzy Sets Syst.*, 82:161–166, September 1996. ISSN 0165-0114. doi: 10.1016/0165-0114(95)00252-9. URL <http://dl.acm.org/citation.cfm?id=241822.241827>.
- [271] Yaochu Jin. Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement. *Fuzzy Systems, IEEE Transactions on*, 8(2): 212–221, apr 2000. ISSN 1063-6706. doi: 10.1109/91.842154.
- [272] M. Setnes, R. Babuška, and H. B. Verbruggen. Complexity reduction in fuzzy modeling. *Math. Comput. Simul.*, 46:507–516, June 1998. ISSN 0378-4754. doi: 10.1016/S0378-4754(98)00079-2. URL <http://dl.acm.org/citation.cfm?id=284142.284163>.
- [273] Timothy J. Ross. *Fuzzy Logic with Engineering Applications*. John Wiley & Sons, August . ISBN 0470860758.
- [274] J. Yen and Liang Wang. Simplifying fuzzy rule-based models using orthogonal transformation methods. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 29(1):13–24, feb 1999. ISSN 1083-4419. doi: 10.1109/3477.740162.
- [275] Yeung Yam. Fuzzy approximation via grid point sampling and singular value decomposition. *Trans. Sys. Man Cyber. Part B*, 27(6):933–951, December 1997. ISSN 1083-4419. doi: 10.1109/3477.650055. URL <http://dx.doi.org/10.1109/3477.650055>.
- [276] Dan Simon. Design and rule base reduction of a fuzzy filter for the estimation of motor currents. *Int. J. Approx. Reasoning*, 25(2):145–167, 2000.

- [277] P. Baranyi and Yeung Yam. Singular value-based approximation with takagi-sugeno type fuzzy rule base. In *Fuzzy Systems, 1997., Proceedings of the Sixth IEEE International Conference on*, volume 1, pages 265–270 vol.1, Jul 1997. doi: 10.1109/FUZZY.1997.616379.
- [278] O. Ciftcioglu. Studies on the complexity reduction with orthogonal transformation. In *Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on*, volume 2, pages 1476 –1481, 2002. doi: 10.1109/FUZZ.2002.1006724.
- [279] O. Kaynak, K. Jezernik, and Agnes Szeghegyi. Complexity reduction of rule based models: a survey. In *Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on*, volume 2, pages 1216–1221, 2002. doi: 10.1109/FUZZ.2002.1006677.
- [280] M. Ramirez-Gonzalez and O.P. Malik. Simplified fuzzy logic controller and its application as a power system stabilizer. In *Intelligent System Applications to Power Systems, 2009. ISAP '09. 15th International Conference on*, pages 1–6, Nov 2009. doi: 10.1109/ISAP.2009.5352817.
- [281] Abdelhay A. Sallam and Om P. Malik. *Power System Stability: Modelling, Analysis and Control*. Power and Energy Series. Institution of Engineering and Technology, 2015. ISBN 978-1-84919-944-5. URL <http://www.theiet.org/resources/books/pow-en/powersys.cfm>.
- [282] J.C. Deville. Méthodes statistiques et numériques de l'analyse harmonique. *Annales de l'insée*, (15):3, 5–101, Jan.-Apr. 1974. ISSN 00190209.
- [283] J.K. Gruber, C. Bordons, R. Bars, and R. Haber. Nonlinear predictive control of smooth nonlinear systems based on volterra models. application to a pilot plant. *International Journal of Robust and Nonlinear Control*, 20(16): 1817–1835, 2010. ISSN 1099-1239. doi: 10.1002/rnc.1549. URL <http://dx.doi.org/10.1002/rnc.1549>.
- [284] Michael Piovoso and Karlene Hoo. The use of multivariate statistics in process control. In *The Control Handbook, Second Edition*, pages 16–1–16–20. Informa UK Limited, dec 2010. doi: 10.1201/b10382-20. URL <http://dx.doi.org/10.1201/b10382-20>.
- [285] Ricardo Dunia and S. Joe Qin. Joint diagnosis of process and sensor faults using principal component analysis. *Control Engineering Practice*, 6(4):457 –

- 469, 1998. ISSN 0967-0661. doi: [http://dx.doi.org/10.1016/S0967-0661\(98\)00027-6](http://dx.doi.org/10.1016/S0967-0661(98)00027-6). URL <http://www.sciencedirect.com/science/article/pii/S0967066198000276>.
- [286] Pradeep B. Deshpande. *Multivariable process control*. Research Triangle Park, N.C. : Instrument Society of America, 1989. ISBN 1556170068. Includes bibliographical references.
- [287] L.B. Palma and F. Vieira Coito. Tuning pca controllers based on manual control data. In *Emerging Technologies Factory Automation (ETFA), 2011 IEEE 16th Conference on*, pages 1–4, Sept 2011. doi: 10.1109/ETFA.2011.6059216.
- [288] L. Brito Palma, F. Vieira Coito, P. Sousa Gil, and R. Neves-Silva. Process control based on pca models. In *Emerging Technologies and Factory Automation (ETFA), 2010 IEEE Conference on*, pages 1–4, Sept 2010. doi: 10.1109/ETFA.2010.5641080.
- [289] Jianguo Yao, Xue Liu, and Xiaoyun Zhu. Reduced dimension control based on online recursive principal component analysis. In *American Control Conference, 2009. ACC '09.*, pages 5713–5718, June 2009. doi: 10.1109/ACC.2009.5160376.
- [290] Petter Tøndel, Tor Arne Johansen, and Alberto Bemporad. An algorithm for multi-parametric quadratic programming and explicit {MPC} solutions. *Automatica*, 39(3):489 – 497, 2003. ISSN 0005-1098. doi: [http://dx.doi.org/10.1016/S0005-1098\(02\)00250-9](http://dx.doi.org/10.1016/S0005-1098(02)00250-9). URL <http://www.sciencedirect.com/science/article/pii/S0005109802002509>.
- [291] A. Bemporad and C. Filippi. Suboptimal explicit mpc via approximate multi-parametric quadratic programming. In *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, volume 5, pages 4851–4856 vol.5, 2001. doi: 10.1109/.2001.980975.
- [292] P. Tondel, T.A. Johansen, and A. Bemporad. Computation and approximation of piecewise affine control laws via binary search trees. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 3, pages 3144–3149 vol.3, Dec 2002. doi: 10.1109/CDC.2002.1184353.
- [293] T.A. Johansen. Reduced explicit constrained linear quadratic regulators. *Automatic Control, IEEE Transactions on*, 48(5):823–829, May 2003. ISSN 0018-9286. doi: 10.1109/TAC.2003.811261.

- [294] S. Skogestad, M. Morari, and J.C. Doyle. Robust control of ill-conditioned plants: high-purity distillation. *Automatic Control, IEEE Transactions on*, 33 (12):1092–1105, Dec 1988. ISSN 0018-9286. doi: 10.1109/9.14431.
- [295] Samira Roshany-Yamchi. *A Distributed Model Predictive Control For Multi-Agent Multi-Rate Systems*. PhD thesis, Cork Institute of Technology, Rossa Avenue, Bishopstown. Cork, 2012.
- [296] J.M. Escaño and C. Bordons. Complexity reduction in fuzzy systems using functional principal component analysis. In Fernando Matía, G. Nicolás Marichal, and Emilio Jiménez, editors, *Fuzzy Modeling and Control: Theory and Applications*, volume 9 of *Atlantis Computational Intelligence Systems*, pages 49–65. Atlantis Press, 2014. ISBN 978-94-6239-081-2. doi: 10.2991/978-94-6239-082-9_3. URL http://dx.doi.org/10.2991/978-94-6239-082-9_3.
- [297] W. Pedrycz. *Fuzzy Modelling: Paradigms and Practice*. International Series in Intelligent Technologies. ISBN 9781461313656.
- [298] J.M. Escaño, C. Bordons, M.R. Garcia, C. Vilas, and A.A. Alonso. Control predictivo basado en modelo neuroborroso de un autoclave industrial. In *XXVIII Jornadas de Automática. Huelva. Universidad de Huelva*, 2007.
- [299] J.M. Escaño, C. Bordons, and A. Nuevo. Dynamical & non-dynamical neuro-fuzzy models of a mixing chamber pressure. In *2008 SICE Annual Conference*, 2008.
- [300] J.M. Escaño and C. Bordons. Neurofuzzy model of an industrial processs, reducing complexity by using principal component analysis. In *XVI Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF 2012)*., 2012.
- [301] J.M. Escaño, D. Algarín, and M.G. Ortega. Identificación y control de posición de un sistema de levitación neumática. In *XXV Jornadas de Automática. Universidad Castilla la Mancha*. 2004. ISBN: 84-688-7460-4, 2004.