University of Coimbra
Faculty of Sciences and Technology
Department of Electrical and Computer Engineering

Saeid Rastegar

Multi Parametric Intelligent Identification and Robust Control Methodologies For Industrial Processes

Tese de Doutoramento em Engenharia Electrotécnica e de Computadores, ramo de especialização em Automação e Robótica, orientada pelo Prof. Dr. Rui Alexandre de Matos Araújo e apresentada ao Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

January 2017

Universidade de Coimbra

· U  C ·

University of Coimbra
Faculty of Sciences and Technology
Department of Electrical and Computer Engineering

Multi Parametric Intelligent Identification and Robust Control Methodologies For Industrial Processes

by

Saeid Rastegar

Tese de Doutoramento em Engenharia Electrotécnica e de Computadores, ramo de especialização em Automação e Robótica, orientada pelo Prof. Dr. Rui Alexandre de Matos Araújo e apresentada ao Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Coimbra
January 2017

Acknowledgements

I would like to start by acknowledging my advisor, Prof. Rui Alexandre de Matos Araújo, who constantly listened to my some very scraped thoughts and guiding them into the right direction. His advice and his foresight kept me focus on the main objectives, and at the same time he gave me enough hope to move towards the future. Throughout my studies he gave me enough motivation when I couldn't see any light at the end. Thank you for all the support and positive energy you gave me.

A special thanks to my family. Words cannot express how grateful I am to my mother, father, and my two brothers for all of the sacrifices that you have made on my behalf. Your prayer which arisen from your faith for me was an important support for me.

The Control Lab at the "Institute for Systems and Robotics - University of Coimbra" (ISR-UC) has been my home for the last five years, and the members there made my second family. Everybody there had a role in my understanding of control theories. Thank you for all of them. Alireza Emami, no words can describe how happy I am for having a good friend like you. Jérôme Mendes, thank you for helping me in fuzzy control.

I thank the Foundation for Science and Technology (FCT) that supported this thesis through the doctoral fellowship with reference SFRH/BD/89186/2012, that was supported by the "Human Potential Operational Programme" (POPH) and by the "European Social Fund" (ESF). I thank Project "Self-Learning Industrial Control Systems Through Process Data" (SCIAD), with reference SCIAD/2011/21531, co-financed by "National Strategic Reference Framework" (QREN), in the framework of the "Mais Centro - Regional Operational Program of the Centro", and by the European Union through the "European Regional Development Fund" (ERDF).

Finally, I would like to thank all friends for making my stay in Coimbra, Portugal,

an excellent stay; Whether close friends who are in my mind at this time or those people who inspire me great moments whenever I think to them.

Abstract

Focus on creative autonomous intelligent systems for industrial control applications is an absolutely crucial aspect in the recent decades. There are enormous economic incentives to optimize industrial control processes. The World's evolution from traditional old factory layouts to modern industrial designs increased the global community attention to environmental pollution reduction and energy consumption reduction. In industry, large attention to the energy dissipation concern increased the vast worldwide competition in applicable control techniques to achieve adequate, high-quality, and optimal control designs for industrial processes. However, normally the implementation of a good, or an optimal, control strategy for industrial processes is not an easy task. Frequently, systems have complex characteristics such as nonlinearities, unknown and time-varying dynamics, constraints, disturbances, and uncertainties. A vast amount of researches have already been performed in both theoretical and practical control aspects. However, in the near future higher levels of accuracy, flexibility, and reliability will be required for industrial plants.

This thesis will address both identification and control problems on nonlinear industrial processes using fuzzy logic theory, and model predictive control. Also, the design of a robust control framework for uncertain systems will be addressed in this thesis. The development of control designs which can deal with concerns such as modeling difficulties, uncertain dynamics, time-varying parameters, constraints, and disturbances, is an important topic which will be approached in this thesis. In this context, four main research objectives and research directions are considered.

The first objective is to propose an automatic algorithm to identify the Takagi-Sugeno (T-S) fuzzy model and its parameters for modeling nonlinear processes by using a numerical data set of input/output data of the process and using fuzzy logic systems and optimization algorithms. Two different approaches to identify the T-S fuzzy model parameters are investigated. In the first approach, a new unsuper-

vised fuzzy clustering algorithm (NUFCA), integrated with a recursive least squares (RLS) algorithm, is proposed to construct a novel methodology for online evolving T-S fuzzy model identification. In the second approach, a novel identification methodology based on a hierarchical particle swarm optimization (HPSO) approach is introduced to model unknown nonlinear processes from a set of input/output data by automatically extracting the fuzzy logic system (FLS)'s parameters of the T-S fuzzy model. During online operation, both the consequent parameters of the T-S fuzzy model and the particle swarm optimization (PSO) inertia weight are continually updated by a self-adaptive HPSO (S-AHPSO) algorithm.

The second objective of the thesis is to use of the T-S fuzzy models which become available as results of the first objective, to construct a fuzzy model predictive control architecture for nonlinear time-varying systems without initial knowledge about the mathematical model of the plant. The integration of each T-S identification methodology proposed in the work related to the first objective with the generalized predictive control (GPC) results in an efficient adaptive fuzzy generalized predictive control methodology (AFGPC).

The third objective of the thesis is devoted to the development of a new robust constrained control methodology for discrete-time linear parameter varying (DT-LPV) uncertain systems, based on a synergetic control theory (SCT) approach. Moreover, given an uncertain DT-LPV system jointly subject to unmeasured and constrained additive disturbances, and constraints in states, input commands, and reference signals (setpoints), then invariant set theory is used to find an appropriate polyhedral robust invariant region in which the proposed control framework is guaranteed to robustly stabilize the closed-loop system.

Finally, the fourth objective of the thesis is to design a controller based on a proportional-integral (PI) type of SCT macro-variable to control a non-linear state dependent discrete time varying (NSDDTV) dynamical systems.

To validate and demonstrate the robustness and efficiency of the proposed methodologies, they are applied to identify and control the dissolved oxygen concentration in an activated sludge reactor within a wastewater treatment plant (WWTP), and a real experimental setup composed of two coupled DC motors; to identify a real WWTP plant; and to control a simulated continuous stirred tank reactor (CSTR), and a simulated process of the Escherichia Coli (E. Coli) bacteria culture.

Resumo

O foco em sistemas inteligentes autónomos criativos para aplicações de controlo industrial é um aspecto absolutamente crucial nas últimas décadas. Existem enormes incentivos económicos para optimizar os processos de controlo industrial. A evolução mundial de layouts tradicionais em fábricas antigas para industrias modernas aumentou a atenção da comunidade global à poluição ambiental e ao consumo energético. Na indústria, a elevada preocupação com a dissipação/utilização de energia aumentou a competição sobre técnicas de controlo aplicáveis de modo a alcançar projectos de controlo adequados e de alta qualidade para processos industriais. Contudo, normalmente a implementação de uma boa estratégia, ou uma estratégia óptima, de controlo para processos industriais não é uma tarefa fácil. Frequentemente, os sistemas têm características complexas tais como não linearidades, dinâmicas desconhecidas e variáveis no tempo, restrições, perturbações e incertezas. Muitas pesquisas têm sido feitas em aspectos teóricos e práticos de controlo. No entanto, no futuro próximo serão necessários níveis mais elevados de precisão, flexibilidade e confiabilidade para plantas industriais.

Esta tese irá abordar ambos os problemas de identificação e controlo em processos industriais não-lineares usando a teoria de lógica difusa e modelos de controlo preditivo. Adicionalmente será abordado nesta tese o projecto de uma estrutura de controlo robusto de sistemas incertos. O controlo de processos não-lineares, sujeitos a fatores tais como dificuldade de modelação, dinâmicas incertas, parâmetros variáveis no tempo, restrições e perturbações é um tópico importante que será abordado nesta tese. Neste contexto, quatro principais objetivos de investigação e direções de investigação são considerados.

O primeiro objectivo é propor um algoritmo automático para identificar um modelo difuso de Takagi Sugeno (T-S) e os seus parâmetros, para modelação de processos não-lineares, utilizando um conjunto de dados numéricos de entrada/saída do

processo, e usando sistemas de lógica difusa e algoritmos de optimização. São investigadas e usadas duas abordagens diferentes para identificar os parâmetros do modelo difuso T-S. Primeiro, um novo algoritmo de aglomeração não-supervisionado, integrado com um método de mínimos quadrados recursivo, é proposto para construir uma nova metodologia para identificação evolutiva e online de um modelo difuso de T-S. Na segunda abordagem, uma nova metodologia de identificação baseada numa abordagem de optimização hierárquica por enxame de partículas é introduzida para modelar processos não-lineares desconhecidos a partir de um conjunto de dados de entrada/saída através da extracção automática dos parâmetros do sistema de lógica difusa do modelo difuso T-S. Durante a operação online, tanto os parâmetros consequentes do modelo difuso T-S como o peso de inércia do processo de optimização por enxame de partículas, são continuamente atualizados por um algoritmo auto-adaptativo hierárquico por enxame de partículas.

O segundo objetivo desta tese, é a utilização dos modelos difusos T-S que se tornam disponíveis como resultado do primeiro objectivo desta tese, para construir uma arquitectura de controlo predictivo baseada em modelos difusos para sistemas não-lineares variantes no tempo, sem conhecimento inicial do modelo matemático da planta. A integração de cada metodologia de identificação de T-S proposta no trabalho relacionado com o primeiro objetivo, com a metodologia de controlo predictivo generalizado, resulta numa eficiente metodologia de controlo preditivo adaptativo difuso generalizado.

O terceiro objetivo da tese é dedicado ao desenvolvimento de uma nova metodologia de controlo robusto sujeito a restrições baseada na teoria de controlo sinergético (SCT), para sistemas discretos lineares e variantes no tempo (DL-VT) contendo incertezas. Além disso, dado um sistema incerto DL-VT conjuntamente sujeito a perturbações aditivas sujeitas a restrições e não-mensuráveis, e restrições nos estados, comandos de actuação e sinais de referência (*setpoints*), é usada a teoria de conjuntos invariantes para encontrar uma adequada região robusta invariante poliédrica, na qual é garantido que a estrutura de controlo proposta estabiliza robustamente o sistema em malha fechado.

Finalmente, o quarto objectivo da tese é o projecto de um controlador baseado numa macro variável SCT do tipo proporcional-integral (PI) para controlar sistemas dinâmicos em tempo discreto não lineares, dependentes do estado, e variantes no tempo (NSDDTV).

Para validar e demonstrar a robustez e eficiência das metodologias propostas, estas são aplicadas para identificar e controlar a concentração de oxigénio dissolvido num reator de lodo activado, dentro de uma planta de tratamento de águas residuais, e um ambiente experimental real, composto por dois motores DC acoplados; para identificar uma planta real de tratamento de águas residuais; e para controlar um reactor contínuo do tipo tanque agitado simulado, e controlar um processo simulado de culturas de bactérias Escherichia Coli (E. Coli).

Abbreviations and Symbols

General Abbreviations

ADF	Adaptive Directional Forgetting
AI	Adjacency Index
AFGPC	Adaptive Fuzzy Generalized Predictive Control
APSO	Adaptive Particle Swarm Optimization
ARX	Auto-Regressive eXogenous
BSM1	Benchmark Simulation Model 1
BTP	Burn-Through Point
CFTOC	Constrained Finite-time Optimal Control
CPR	Counts Per Revolution
CSTR	Continuous Stirred Tank Reactor
DAFC	Direct Adaptive Fuzzy Control
DFC	Direct Fuzzy Control
DFIG	Double-Fed Induction Generator
DL-VT	Discreto Linear e Variante no Tempo
DMC	Dynamic Matrix Control
DT-LPV	Discrete Time Linear Parametric Varying
EC	External Carbon
ECG	ElectroCardioGram
E. Coli	Escherichia Coli
EEG	ElectroEncephaloGram
FCM	Fuzzy C-Means
FIE	Fuzzy Inference Engine

FIR	Finite Impulse Response
FLC	Fuzzy Logic Control
FLS	Fuzzy Logic System
FRLS	Forgetting Recursive Least Squares
GAs	Genetic Algorithms
GPC	Generalized Predictive Control
HGA	Hierarchical Genetic Algorithm
HIECON	Hierarchical Constraint Control
HIV-1	Human Immunodeficiency Virus Type 1
HPSO	Hierarchical Particle Swarm Optimization
IAFC	Indirect Adaptive Fuzzy Control
IDCOM	Identification and Command
ILLSA	Incremental Local Learning Soft Sensing Algorithm
IRC	Intersection Rule Configuration
IRIS	Integrated Risk Information System
LDW-PSO	Linearly Decreasing Inertia Weight PSO
LMI	Linear Matrix Inequality
LP	Linear Programming
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
MCS	Maximum Constrains Satisfaction
m-GA	Micro-Genetic algorithm
MHE	Moving Horizon Estimation
MIMO	Multiple Input Multiple Output
MLPNN	Multilayer Perceptron Neural Networks
MPHC	Model Predictive Heuristic Control
MR	Magneto Rheological
MRI	Magnetic Resonance Imaging
MSE	Mean Square Error

NFCRMA	New Fuzzy C-Regression Model Algorithm
NN	Neural Network
NSDDTV	Non-linear State Dependent Discrete Time Varying
NUFCA	New Unsupervised Fuzzy Clustering Algorithm
PC	Partition Coefficient
PCT	Predictive Control Technology
PFC	Predictive functional Control
PI	Proportional-Integral
PSC-SEA	Partial Solutions Consideration based a Self-adaptive Evolutionary Algorithm
PSO	Particle Swarm Optimization
QDMC	Quadratic Dynamic Matrix Control
RDSCTC	Robust Discrete Synergetic Control Theory Control
RDSCTC-PI	Robust Discrete Synergetic Control Theory Control of Proportional-Integral Type
RLS	Recursive Least Squares
RLS-ADF	Recursive Least Squares-Adaptive Directional Forgetting
RMMB	Robust Macro-variable Manifold Band
RMPC	Robust Model Predictive Controller
RMPCT	Robust Model Predictive Control Technology
RPC	Repetitive Predictive Control
RPLS	Recursive Partial Least Squares
S-AHPSO	Self-Adaptive Hierarchical Particle Swarm Optimization
SC	Synergetic Control
SCT	Synergetic Control Theory
SDSMC	Standard Discrete-time Sliding Mode Controller
SISO	Single Input Single Output
SMC	Sliding Mode Control
SMCA	Setpoint Multivariable Control Architecture
SMOC	Shell Multivariable Optimizing Controller
SPSO	Standard Particle Swarm Optimization
T-S	Takagi-Sugeno

TSFM	Takagi-Sugeno Fuzzy Model
UKF	Unscented Kalman Filter
URC	Union Rule Configuration
WT	Wind Turbines
WWTP	Waste Water Treatment Plant

General Symbols

$\boldsymbol{\theta}_i$	Parameter vector with adjustable parameters of the i-th fuzzy rule
$\boldsymbol{\Theta}$	Matrix that contain all adjustable parameter vector $\boldsymbol{\theta}_i$
θ_{ij}	Adjustable parameter of the i-th fuzzy rule and of the input variable j
γ	Positive constant
β	Positive constant
c_{1r}	First positive acceleration coefficient of r-th particle
c_{2r}	Second positive acceleration coefficient of r-th particle
$\mathbf{I}$	Identity matrix
r_{max}	Maximum number of particles in each level
R_i	i-th fuzzy rule
T_2	Maximum number of S-AHPSO iterations
$\mathbf{v}_{min}^p$	Minimum velocity in Level p
$\mathbf{v}_{max}^p$	Maximum velocity in Level p
$w_{r,min}$	Minimum inertia weight of particle r for the S-AHPSO Algorithm
$w_{r,max}$	Maximum inertia weight of particle r for the S-AHPSO Algorithm

Takagi-Sugeno Fuzzy Model Identification (Chapter 4)

γ_i	Parameter of the i-th fuzzy rule of RLS method
ν_i	Parameter of the i-th fuzzy rule of RLS method
ρ	Positive constant parameter of RLS method
σ_{ij}	Gaussian dispersion of membership functions i of the variable j
τ_i	Parameter of the i-th fuzzy rule of RLS method
φ_i	Forgetting factor of the fuzzy rule i of RLS method

μ_i	Fuzzy partition of fuzzy subsets i
$\mu_{A_j^i}$	Antecedent fuzzy membership functions of the i-th fuzzy rule of the variable j
η	Fuzziness weighting parameter
γ	positive constant
Δ	Difference operator
$\delta(k)$	Kronecker Delta function
$\mathbf{\Theta}_{on}$	Adaptive consequent parameter
a	Number of particle at Level 6
a_{max}	Maximum number of particles at Level 6
A^i	i-th fuzzy subsets
b	Number of particle at Level 5
b_{max}	Maximum number of particles at Level 5
c	Number of cluster
c^*	Optimal clustering number
c_1	Random number
c_2	Random number
$\mathbf{C}_i$	Covariance matrix of the i-th fuzzy rule
d	Number of particle at Level 4
d_{il}	Eucliden distance between the observation $\mathbf{x}_l$ and the cluster centroid $\mathbf{v}_i$
d_{max}	Maximum number of particles at Level 4
E_i	i-th auxiliary set of samples
E_i^*	Optimal i- th set of samples
$\mathbf{g}_i$	Center of E_i
$\mathbf{gbest}^p(t)$	Best previous position attained among all particles of the swarm, in each level of p, until the current iteration t
$\mathbf{gbest}(t)$	Best previous position attained among all particles of the swarm, for the S-AHPSO Algorithm
i	i-th fuzzy rule
j	j-th input variable
J	Objective function
J_3^z	Fitness function of z-th particle at Level 3
J_4^d	Fitness function of d-th particle at Level 4

J_6^a	Fitness function of a-th particle at Level 6
J_e^u	Fitness function of u-th particle at Level e
K	Number of K nearest neighbourhood (KNN)
l	An observation from data
L	Number of observations
m	Number of particle at Level 1
m_{max}	Maximum number of particles at Level 1
N	Number of input variables
p	Number of level
$\mathbf{pbest}_r^p(t)$	Best previous position of the r-th particle in p- level, for all non-future values of t
$\mathbf{pbest}_r(t)$	Best previous position of the r-th particle, for all non-future values of t, for the S-AHPSO Algorithm
r	Particle number
s	Number of particle at Level 2
s_{max}	Maximum number of particles at Level 2
t	Iteration number
T_1	maximum number of the HPSO iterations
u_{il}	i-th row of the matrix $\mathbf{U}$ at l-th observation
$\mathbf{U}$	Fuzzy partition matrix
$\mathbf{v}_i$	Cluster centroid vectors
v_{ij}	Center of the membership function i of the input variable j
$\mathbf{v}_r^p(t)$	Velocity of particle r at t-th iteration in p-th level
v_t	Clustering validity index
v_t^*	Optimal clustering validity index
$\mathbf{V}$	Matrix of cluster centroid vectors
w	Inertia weight
$\mathbf{w}_r^p(t)$	Interia weight of r-th particle of p-th level at t-th iteration
$w_{r,max}^p$	Maximum inertia weight of particle r in Level p
$w_{r,min}^p$	Minimum inertia weight of particle r in Level p
$\mathbf{x}$	Input variable vector
$\mathbf{x}^i$	i-th unknown sample which is farthest from $\mathbf{g}_{i-1}$

x_j	Input variable j of the fuzzy system
x_{lj}	j-th input variable from the l-th observation
$\mathbf{x}_r^p(t)$	Position of particle r at t-th iteration in p-th level
$\mathbf{X}$	Data matrix
y_l	The l-th output observation
$y(l)$	The l-th output observation
$\hat{y}_l$	The corresponding output value predicted by the FLS
$y^{(\mathbf{gbest})}(t)$	Best output estimation attained among all particles of the swarm, until the current iteration t
$y^{(\mathbf{pbest}_r)}(t)$	Best output estimation attained by the r-th particle for all non-future values of t
$\hat{y}^{(r)}(k)$	Prediction of $y_{re}(k)$ calculated by the r-th particle
$y_{re}(k)$	Target output value at instant k
z	Number of particle at Level 3
z_{max}	Maximum number of particles at Level 3

Adaptive Fuzzy Model Predictive Control (Chapter 5)

$\lambda(z^{-1})$	Weighting polynomial
A_{ij}	Linguistic term characterized by fuzzy membership functions $\mu_{A_j^i}(x_j)$
$\bar{d}$	Time-delay of the system
$DO_{ref}(t)$	Setpoints of BSM1 plant
n_u	Input order
n_y	Output order
N_p	Output horizon
N_u	Control horizon
p	p-step ahead prediction
$r(k+p)$	An p-step of the future reference trajectory at k-th sample time
$r(k+p)$	Future reference trajectory
$R_L(k)$	Load disturbance
T_n	Total number of sample
$u(\cdot)$	Process input
$u(k+p)$	An p-step ahead prediction of the controller

$y(\cdot)$	Process output
$\hat{y}(k+p)$	An p-step ahead prediction of the system.
$z_d(k)$	Setpoints of DC motors setup

Robust Control Design Based on Synergetic Control Theory (Chapter 6)

$\|\| \cdot \|\|$	Euclidean distance
$\alpha(k)$	Uncertain parameter
$\beta(k)$	Uncertain parameter
$\boldsymbol{\psi}(k)$	Vector of aggregated macro-variables
$\Gamma \cap \Phi$	Intersection of convex polyhedra Γ and Φ
$Co\{\cdot\}$	Convex hull of a set
ψ_i	i-th aggregated macro-variables
$Proj_X(\Gamma)$	Projection of Γ onto X
ω	Angular velocity of the motor 1 and motor 2
μ_i	Specific growth rate
$\zeta(t)$	Internal disturbance
$\mathbb{R}$	Field of real numbers
$\mathbb{R}^n$	n-dimensional Euclidean space
$\bar{\boldsymbol{\varrho}}$	A set represents nominal system matrices in state-space domain
$\boldsymbol{\varrho}$	A set represents real system matrices in state-space domain
Ξ	A constraints set on the plant state
$\mathcal{J}$	A polyhedron
$\vartheta(t)$	External disturbance
$\mathtt{X}_0$	Projection of $\mathcal{O}_\infty$
Ξ	Constraint admissible set
A	Acetate
A_{ref}	Setpoints of Acetate
$\bar{\mathbf{A}}_1$, $\bar{\mathbf{A}}_4$	Two $1 \times n$ matrices
$\bar{\mathbf{A}}_2$, $\bar{\mathbf{A}}_3$	Scalar values
$A \sim B$	Pontryagin difference of two polyhedra $A \subset \mathbb{R}^n$ and $B \subset \mathbb{R}^n$

$A \oplus B$	The Minkowski sum of convex polyhedra A and B
$(\mathbf{A},\mathbf{B},\mathbf{C})$	Real system matrices
$(\bar{\mathbf{A}},\bar{\mathbf{B}},\bar{\mathbf{C}})$	Nominal system matrices
$\mathbf{A} * U$	Linear map of a polyhedron, where $\mathbf{A}$ is a real matrix and U is a polyhedron
$(\mathbf{A}_j^v, \mathbf{B}_j^v)$	j-th vertices of the convex hull v
B_{ti}	Viscous friction coefficient
C	Carbon dioxide
C_{tr}	Carbon dioxide transfer rate from liquid to gas phase
abs($\mathbf{E}$)	Matrix of the absolute values of the corresponding elements of matrix $\mathbf{E}$
F_{in}	Influent flow rate
i_i	Armature current
J_{ti}	Moment of inertia
k_i	Stoichiometric coefficients
K_A	Saturation constants of acetate
$K_{i,A}$	Inhibition constants related to acetate uptake
$K_{i,O}$	Inhibition constants related to oxygen uptake
K_O	Saturation constants of oxygen
K_{OS}	Oxygen yield related to glucose
K_S	Saturation constants of substrate (glucose)
K_{ti}	Torque constant of motor i
K_{vi}	Velocity constant
L_{ti}	Inductance
N	Control horizon
O	Oxygen
O_{tr}	Oxygen transfer rate from gas to liquid phase
$\mathcal{O}_\infty$	Robust control invariant set
q_{AC}	Product oxidative rate
$q_{AC,max}$	Maximum of product oxidative rate
q_O	Respiratory capacity
$q_{O,max}$	Maximum of respiratory capacity

q_S	Kinetic term associated with the glucose consumption
$q_{S,crit}$	Critical substrate consumption
$q_{S,max}$	Maximum of term associated with the glucose consumption
q, t, s	Positive integers
r_j	Growth rates
R	Symmetric and positive definite matrix
$R_L(k)$	Load disturbance
R_{ti}	Armature resistance
S	Glucose (substrate)
S_{in}	Influent glucose concentration
T	Convergence rate
T_s	Sample time
$\mathbf{u}(k)$	Control signal
$\mathcal{U}$	Constraint sets (polyhedral sets) in the input signal
$\|\|\mathbf{v}\|\|_\infty$	Chebyshev vector norm of vector $\mathbf{v}$
V_1	voltage source of motor 1
$V(k)$	Lyapunov function
$\mathbf{w}(k)$	A bounded, and unmeasured disturbance
$\bar{\mathbf{w}}(k)$	One-step delayed estimation of $\mathbf{w}(k)$
$\mathbf{w}_{max}$	Row-wise maximum of $\mathbf{w}(k)$
$\mathbf{w}_{min}$	Row-wise minimum of $\mathbf{w}(k)$
W	Culture medium weight
$\mathcal{W}$	Constraint sets (polyhedral sets) in the disturbance
$\mathbf{x}(k)$	System variables
X	Biomass
$\mathcal{X}$	Constraint sets (polyhedral sets) in the state
$\mathbf{y}(k)$	Output variable
$\mathbf{z}_d$	Setpoints variable (Reference variable)
$\mathcal{Z}$	Constraint sets (polyhedral sets) in the setpoints

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Contents

1.1 Main Motivation

In the past twenty years, technology has changed the nature of manufacturing. Not very long time ago, manufacturing and fabrication were all done by human sources. After that, the concept of process automation was gradually invented and then developed. Reduction in production time, increase in accuracy and reliability, less human error, and increase in safety, are the main initial purposes which were starting to be sought behind these technological activities. Very soon, steps to advance industrial

control systems technology started in order to respond to industrial demands for better performance of systems.

Improving the efficiency in industries needs research and investments in the control field in order to create significant financial returns. It is quite frequent in processes in industrial environments to see complex characteristics such as nonlinearities, unknown and time-varying dynamics, constraints, disturbances, and uncertainties. Indeed, a vast amount of researches has been performed in both theoretical and practical aspects of control. However, more advances are required to make a safe, robust, optimal, and efficient control performance with changes in features or structures which can be expected the industrial processes.

Motivated by the aforementioned concerns, several works with model predictive control (MPC), fuzzy logic system (FLS) for nonlinear industrial systems, and robust control design for uncertain system processes have been recently developed. In this context, the first focus of the thesis is to propose methodologies for the control of nonlinear systems, in particular in the areas of FLS and MPC. In continuation, the thesis is devoted to a new robust control framework which uses a synergetic control theory (SCT) approach to control discrete-time linear parameter varying (DT-LPV) systems. The designed controller for DT-LPV systems is called robust discrete SCT controller (RDSCTC). The proposed RDSCTC deals with system uncertainties, persistent unknown disturbances, and constraints. Stability analysis is given, showing that during the operation of the process, and for any unmeasured bounded disturbances, the proposed controller accomplishes the goal of stabilizing the system by asymptotically driving the error of the controlled variable to a bounded SCT macro-variable set containing the origin and then maintaining it there. As another case study, a controller based on a proportional-integral (PI) type of SCT macro-variable is presented. The proposed controller based on a PI-type of SCT macro-variable, is applied to control non-linear state dependent discrete time varying (NSDDTV) dynamical systems.

1.2 Use of Fuzzy Logic Concepts in Control Design Applications

Many control approaches are dependent on models of dynamic plants. An appropriate controller can be designed, if a sufficiently accurate model of the system is obtained. Mostly, in industry, due to complex conditions such as nonlinearity and uncertainty in a system, finding an explicit mathematical form for modelling of the plant, is not an easy task. To overcome this concern, using fuzzy logic theory can be considered as a powerful approach which can be applied to both system identification methodologies and control design applications. Fuzzy logic theory can be explored for system modeling purposes, including for complex plants, being considered a gray-box technique on the boundary between nonlinear black-box and explicit mathematical type of model. Several fuzzy logic concepts exist and their use and integration may depend on the specific application. For example, there are two well-known fuzzy inference engines which make two different fuzzy model types: Mamdani and Takagi-Sugeno (T-S) models. Fuzzy logic systems have the capability to be presented in a standard form or in hierarchical structures. Each type of structure has its own advantages. Normally, for simple fuzzy applications on a plant with less complexity, a standard form of fuzzy logic structure can give an acceptable performance to address the main control objective. However, this efficiency can be violated when the system is exposed to an increasing number of plant inputs, and exhaustive membership functions, and consequently large numbers of potential fuzzy rules numbers. In this condition, maybe a hierarchical fuzzy structure can be an appropriate alternative which can help to achieve an efficient identification and/or control design. Furthermore, fuzzy logic system applications in both system identifications and control designs have the capability to be combined with computational intelligence techniques to improve their overall frameworks.

1.3 Industrial MPC Technology Motivation

Generally, the main objective in MPC design is to find a future trajectory of the input manipulated variable in such a way that it can optimize the future behavior of the plant. The optimization procedure is performed in a time window by giving

the current plant status at the start of the time window. There are three general approaches to MPC design where each approach uses its own model structure. The MPC control design which uses a finite impulse response (FIR) model as well as a step response model, the MPC design based on a state space model, and the MPC designs based on a transfer function model. The MPC design based on a transfer function model is applicable to both stable and unstable plants. Among the MPC designs which employ a transfer function model maybe the most typically used is the generalized predictive control (GPC).

In recent years, MPC utilization has been changed drastically, with a large increase in the number of reported applications. Daily growing popularity of MPC can be found in a wide variety of application areas including chemicals, food processing, automotive, and aerospace applications. Among all the MPC works, there can be found several MPC methodologies where the control design is considered for linear plants. The reason for this was that the procedure for identification of linear plants from input/output data is typically simpler. Also, the MPC quadratic optimization problem issue could be easily solved for the linear prediction terms.

MPC is considered to be not a new control technique for linear plants. However, MPC for complex systems with high degree of non-linearity, is a field under current research. Indeed, recently some MPC based methodologies for nonlinear systems have been proposed. But due to complexity in real-time processing conditions, mostly the current MPC methodologies have met difficulties when they were tested on real conditions. A survey about MPC industrial applications with linear and nonlinear models is presented in [Qin and Badgwell, 2003], where the authors found, in 2003, more than 4600 MPC applications, over twice the number in their previous survey in 1997 [Qin and Badgwell, 1997]. Figure 1.1 shows an evolutionary tree for the most significant linear industrial MPC algorithms. A short description of linear MPC products in Figure 1.1 is presented in Table 1.1.

1.4 Robust Control Design Motivation

Mostly a good control performance can be obtained if a good system model of a plant can be estimated. Frequently, in a system identification process, due to complexity in plant behavior, only an uncertain model can be estimated. In other words, an explicit form of the system model is constructed which contains some uncertain

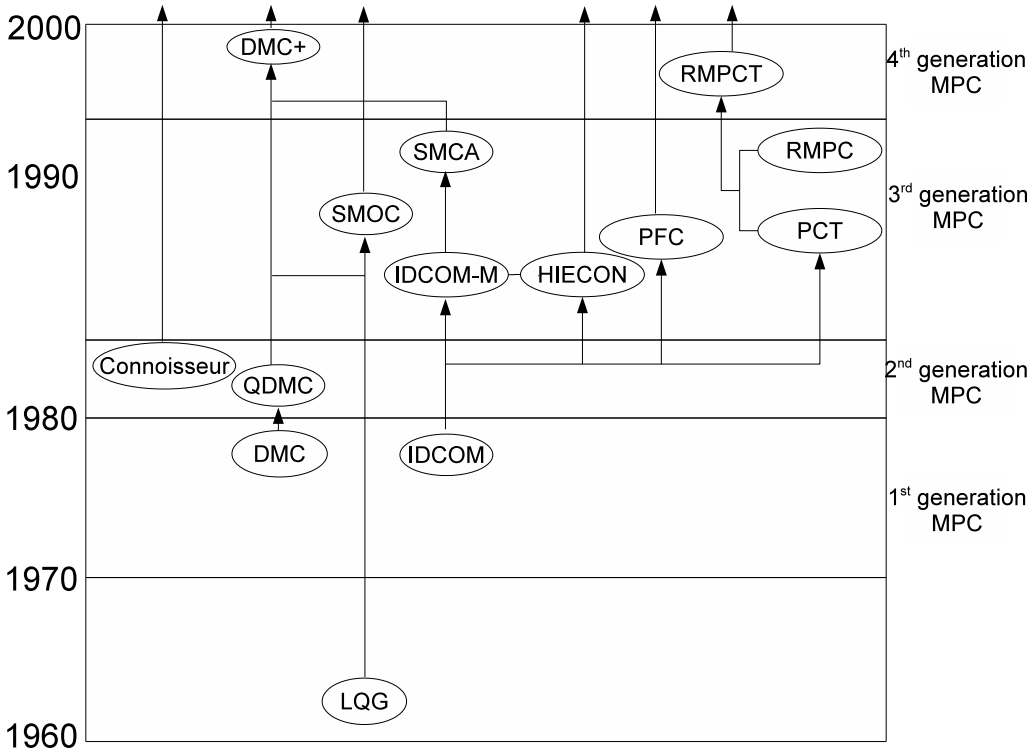


Figure 1.1: Approximate distributed diagram of linear MPC algorithms (from [Qin and Badgwell, 2000]).

parameters in the plant modeling. Also, it happens frequently that in control design, besides having uncertainties inside the estimated model of a plant, also the plant is needed to deal with constraints and disturbances during the control of the process [Rastegar and Araújo, 2013]. Then, an important question that remains is how a control can be designed to deal with this situation. Motivated by these problems, this PhD work will present a new robust constrained control methodology for uncertain systems based on a synergetic control theory (SCT) approach. Also, as another case study, a new control design for a class of non-linear state dependent discrete time varying (NSDDTV) dynamical systems is presented. The proposed control scheme is featured by a proportional-integral (PI) type of synergetic control theory (SCT) macro-variable manifold based on the output error.

In general, SCT provides methods for designing optimal controllers for dynamical systems, where the controllers are coordinated with internal state variables of the systems [Kolesnikov, 2014], [Kolesnikov, 1994, cited in [Kolesnikov, 2014]]. The

Table 1.1: Representative of some commercial linear MPC products/algorithms.

MPC algorithms	Company	Year	Comments
Linear Quadratic Gaussian (LQG)	-	-	It is the combination of a Kalman filter with a linear-quadratic regulator (LQR); constraints on the process inputs, states, and outputs were generally not addressed in the development of LQG theory.
Identification and Command (IDCOM)	Adersa	1973	Also known as model predictive heuristic control (MPHC); input and output constraints are included in the formulation.
Dynamic Matrix Control (DMC)	Aspen Technology	1985	Linear step response model for the plant is needed; future plant output behavior is determined by trying to track the setpoint as closely as possible.
Quadratic DMC (QDMC)	Shell Oil		Second generation of MPC technology, is a form of the DMC algorithm as a quadratic program (QP) in which input and output constraints appear explicitly.
IDCOM-M	Setpoint, Inc	1987	The M was to distinguish this type from a single input/single output version called IDCOM-S, the nearly same Adersa version was referred to as hierarchical constraint control (HIECON).
Shell Multivariable Optimizing Controller (SMOC)	Shell Global Solutions	1998	Third generation of MPC; is a bridge between state-space and MPC algorithms; Kalman filter is used to estimate the plant states and unmeasured disturbances.
Setpoint Multivariable Control Architecture (SMCA)	Setpoint Inc		Third generation of MPC; improved version of the IDCOM-M technology; it provides a natural way to incorporate multiple ranked control objectives and constraints.
Robust Model Predictive Control Technology (RMPCT)	Honeywell	1991	It is representative of the fourth generation MPC technology; contains features such as windows-based graphical user interfaces and multiple optimization levels to address prioritized control objectives.
Predictive Control Technology (PCT)	Honeywell	1984	Covers the topics of steady-state target calculation, infinite horizon receding horizon regulation, resolving infeasibility.
Robust Model Predictive Control (RMPC)	Honeywell	-	Primary version of RMPCT which was merged with the Profimatics PCT controller to create their current offering, i.e. RMPCT.
DMC+	Aspen Tech	-	It is representative of the fourth generation MPC technology; the SMCA and DMC technologies were subsequently merged to create DMC+ product.
Predictive Functional Control (PFC)	Adersa	-	Works with two types of models: realigned and independent. In the independent case the model output may be different from the process output.
Connoisseur	Invensys	-	Works with the following model types: finite impulse response (FIR), auto-regressive with exogenous input (ARX), and multi-model.

resulting dynamical systems with their controllers have areas of attraction that correspond to the control purposes. Depending on the dimensionality of the systems, attractors can be points, contours, tori or regions of fractal dimensionality. If the

requirements that the controllers should provide to the system cannot be fulfilled, the attractors cannot be created, and the dynamical system may become unstable and not converge. Convergence can be analyzed by using stability methods such as the Lyapunov Stability Theory [Kalman and Bertram, 1960; Sastry, 1999; Liu and Hsiao, 2012; Guang-Yue *et al.*, 2013], or using a potential function of the system [Olfati-Saber and Murray, 2002]. In the development of robust control design based on SC theory, concerns such as uncertainties, system constraints, and persistent disturbance are considered.

1.4.1 Invariant Set Theory

Uncertainty in a plant leads to the field of robust control design, with applicability in several areas such as industrial processes. There are several reasons why a plant's control structure may be under an uncertainty situation, such as for example approximate measurement of system model variables, imperfect measurement devices, poor performance in high frequency level, and nonlinearities. As one type of uncertain model representation, all plant uncertainty can be gathered in a convex hull polytope which surrounds the real current/nominal system model. Presence of uncertainty and/or constraints in the system, forces a control design to find a safety working region for the system state variables. In the robust control field, this region is named an 'invariant set'. Set invariance is an important key in the design of control systems when the system is subject to constraints. This theory guarantees that the constraints can be satisfied for all time if and only if the initial state is contained inside an invariant set. The theory of positively invariant sets is used to find a region in which the unconstrained controller does not violate the process constraints. In other words, if all conditions in invariant set theory are satisfied, then for every sampling time k, there is a feasible solution for the control design problem, meeting all system constraints. In most of the cases, due to the complexity in nonlinear systems, finding an invariant set for the related uncertain/constrained system is not easy, but if it is found, it can lead to an efficient robust control design.

1.5 Optimization Algorithms in Fuzzy Control

Direct digital control systems and the computer control of processes has been used to provide a better co-ordinated control and operation in process control and product quality. In computer control of processes, computational intelligence methods are important tools to help in the development of optimal fuzzy control designs. As mentioned before, achieving good performance in MPC is depending on having a process model with good performance. Several types approaches to modeling nonlinear plants can be considered to be used in MPCs. Among them, fuzzy logic based modeling has become an active area of research because of the capability of handling perceptual uncertainties, such as the vagueness and ambiguity involved in the interpretation of a real system. Also, it has shown excellent ability when describing nonlinear systems, in particular with the Takagi-Sugeno (T-S) fuzzy models. Like in many other models where optimization can be used to find parameters, in T-S fuzzy models, optimization algorithms can be used to find the T-S fuzzy model parameters. For example, computational intelligence algorithms such as particle swarm optimization (PSO), neural network (NN) learning algorithms, and genetic algorithms (GAs), individually or in combinational forms have shown a good performance to search for optimal T-S fuzzy model parameters [Coelho and Herrera, 2007; Shuzhi *et al.*, 2012; Mendes, 2014].

1.6 Thesis Contributions

This thesis has the following fundamental contributions:

1. *[Chapter 4], [Rastegar* et al.*, 2016b],[Rastegar* et al.*, 2017b]:* Design of two different novel online evolving Takagi-Sugeno (T-S) fuzzy model identification methodologies for nonlinear systems. The first proposed T-S identification methodology uses a new unsupervised fuzzy clustering algorithm (NUFCA), and a recursive least squares (RLS) method with adaptive directional forgetting (ADF) to construct an online evolving Takagi-Sugeno (T-S) fuzzy model. As the second proposed T-S identification methodology for this chapter, a hierarchical particle swarm optimization (HPSO) algorithm is introduced to automatically extract the structure and all fuzzy logic system (FLS)'s parameters of a T-S fuzzy model by using input/output process data in a way that

it does not require any previous knowledge of the process model, and in particular no previous knowledge concerning the numbers of rules and antecedent fuzzy sets of the T-S fuzzy model. Additionally, the selection of the adequate input variables and their respective time delays for the prediction setting are addressed in the second T-S fuzzy model identification methodology. The proposed HPSO algorithm includes an adaptive procedure and becomes a self-adaptive HPSO (S-AHPSO) algorithm usable in real-time processes.

2. *[Chapter 5], [Rastegar* et al.*, 2016b]:* Design of new methodologies for fuzzy model predictive control of nonlinear time-varying systems without the knowledge about the mathematical model of the plant. The fuzzy systems learned by the proposed T-S fuzzy model identification algorithms in Contribution 1 (see above) are integrated into the control domain to construct an effective adaptive fuzzy generalized predictive control (AFGPC) methodology.

3. *[Chapter 6], [Rastegar* et al.*, 2016a], [Rastegar* et al.*, 2017c], [Rastegar* et al.*, 2017d], [Rastegar* et al.*, 2017a]:* Design of a novel robust constrained control methodology for discrete-time linear parameter varying (DT-LPV) systems is proposed based on a synergetic control theory (SCT) approach, and it is called robust discrete SCT control (RDSCTC). A stability analysis for the RDSCTC is performed and a theorem is given regarding this issue. The design of a robust controller for DT-LPV systems is given, while the systems face uncertainties, and other concerns such as the presence of disturbances on the system, and system constraints. As another case study for this Chapter, a controller based on PI type of the SCT macro-variable which uses a one-step delayed estimation of the disturbance, is designed to control non-linear state dependent discrete time varying (NSDDTV) dynamical systems, while such NSDDTV systems face external disturbances.

1.7 Thesis Organization

The thesis is organized as follows:

1. Chapter 2 presents an overview of fuzzy systems and clustering methodologies.

2. Chapter 3 gives an overview of fuzzy system identification for industrial applications. A description is given concerning the Takagi-Sugeno fuzzy model and its applications. Also, adaptive fuzzy control, and fuzzy predictive control are mentioned in this overview. Finally, constrained robust control design for uncertain systems and synergetic control theory are discussed.

3. Chapter 4 describes the design of two novel online evolving Takagi-Sugeno (T-S) fuzzy model identification methodologies. First, the T-S fuzzy model identification methodology is constructed based on a new unsupervised fuzzy clustering algorithm (NUFCA), and the RLS-ADF method. The second identification methodology utilizes a hierarchical fuzzy logic structure to automatically extract the structure and all T-S fuzzy model parameters by using input/output data in a way that does not require any previous knowledge about the process model.

4. Chapter 5 describes adaptive fuzzy predictive control designs by integration of the T-S fuzzy models learning methodologies proposed in Chapter 4, and model generalized predictive control (GPC) to obtain an effective adaptive fuzzy GPC methodology (AFGPC).

5. Chapter 6 describes a novel robust constrained control methodology for discrete-time linear parameter varying (DT-LPV) systems, that, besides uncertainties, are also subject to other concerns such as the presence of external disturbances, and system constraints. Then, it presents a control design to control non-linear state dependent discrete time varying (NSDDTV) dynamical systems, subject to concerns such as a nonlinearities in the system, and the presence of an external disturbance. Both controllers use a synergetic control theory (SCT) approach to create a robust control strategy.

6. Finally, Chapter 7 presents concluding remarks.

Chapter 2

Overview of Fuzzy Systems and Clustering

Contents

2.1 Clustering Concepts and Analysis

Clustering approaches use information found in the data that describes the objects and their relationships. Clustering methods identify groups of similar observations, and assign items to automatically created groups based on a calculation of the degree of association between items and groups. However, the results can depend on the chosen methods and initial parameter values. The goal is that the objects

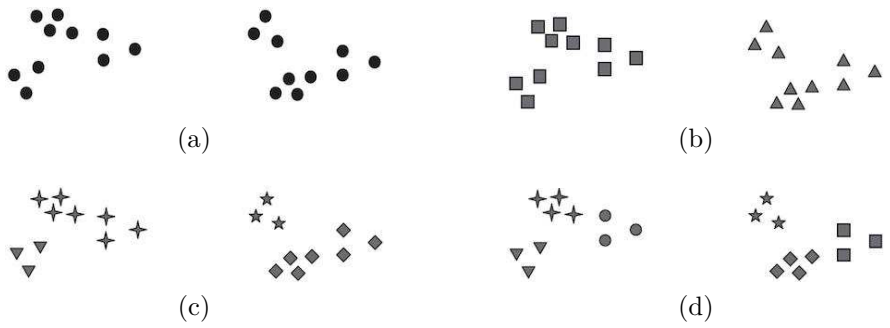


Figure 2.1: Different ways of clustering on the same set of points: (a) original points, (b) two clusters, (c) four clusters, and (d) six clusters.

within a group should be similar to one another, while being different from the objects in other groups. Larger homogeneity within a group, and larger differences between groups, result in a better classification with more distinctiveness in the clustering. Application of clustering techniques can be often found in many fields. Cluster analysis has long been playing an important role with many applications in a wide variety of fields, such as for example in life sciences [Cattell, 2012] (e.g. providing a taxonomy of all living terms such as somatic cells, and gene analysis), categorization in information retrieval [Miyamoto, 2012], analysis of large amounts of climate data [Unal *et al.*, 2003], medicine [Eisen *et al.*, 1998] (e.g. disease clustering of symptom data, and segmentation algorithms in magnetic resonance imaging (MRI), electroencephalogram (EEG), and electrocardiogram (ECG)), and economy [Davies and Walters, 2004] (e.g. market analysis).

2.1.1 Different Types of Clusters

Clustering techniques are applied to conceptually analyze data of objects and divide them into subsets of objects that contain similar characteristics. An example is given, in order illustrate the differences in clusters attributes, as shown in Figure 2.1. In Figure 2.1, the shapes of the markers indicate cluster membership. Figures 2.1(b), 2.1(c), and 2.1(d) divide the data into two, four, and six parts, respectively. Some points can be concluded. Different clusters can show a different classification quality on a same data set depending on the nature of data, on the complexity in the overlapping of clusters, and also on the threshold which can be defined to classify objects. As it can be seen, each of the two groups in Figure 2.1(b) has been divided

into three sub-clusters by another clustering method in Figure 2.1(d).

Not surprisingly, there are several different cluster techniques that have already shown useful performances in practice. Some popular clustering types are as described below:

- **Well-Separated.** Each cluster is a set of members (objects) in which each member has more similar characteristics to every other cluster member than to any member not in the cluster. Sometimes a threshold is considered to classify that all the members (objects) in a cluster must be sufficiently similar to one another. The distance between any two objects in different clusters is larger than the distance between any two objects within a cluster. Well-separated clusters do not need to have a particular distribution and can have any shape;

- **Prototype-Based.** Each cluster is composed of a set of members (objects) in which each member is more similar to the default prototype that defines the cluster than to the prototype of any other cluster. For data with continuous attributes, the prototype of a cluster is often a centroid (the average of all the points in the cluster). However, another situation can happen: a medoid prototype. For example, when the data has categorical attributes, the prototype has often a medoid clustering pattern. A medoid is a representative member of a cluster whose average dissimilarity to all the members in the cluster is minimal. A medoid is a concept similar to a mean or a centroid but a medoid is always a member of the cluster;

- **Graph-Based.** Some times it can be seen that data is represented as a graph. In this case, the nodes are considered as objects and the links represent connections among objects. A cluster is defined as a connected component, i.e. a group (cluster) of objects that are connected to one another, but the objects do not have any connection to the outside of the group. A good example for graph-based classification are contiguity-based clusters, where two objects are connected only if they are within a specified distance of each other. This means that each point in a contiguity-based cluster is closer to some other points in the group. Meanwhile, it is far away from any objects in other clusters;

- **Density-Based.** The basic strategy to distinguish clusters is based on the idea that a cluster in a data space is a contiguous region of high density of

points. The clusters are separated from other clusters by contiguous regions of low point density. The data points in the separating regions of low point density are typically considered noise/outliers;

- **Conceptual Clusters.** Clusters mostly contain a hierarchical structure. A process is developed to construct a conceptual network to characterize a collection of objects. Nodes mark the concepts describing objects clusters, and links mark the relationships between the clusters.

2.1.2 A Visual Example

To show the efficiency of clustering techniques, a clustering example is given in this section. A fuzzy clustering algorithm is applied on five standard flower data sets of integrated risk information system (IRIS) [Fisher, 1936]. The samples of this data set contain real data values with high degree of fuzzy intersection between different clusters. The S_1-S_5 sets contain 2, 3, 4, 6, and 3 overlapping clusters, respectively. The performances of the fuzzy clustering method on the S_1-S_5 data sets are depicted in Figure 2.2. The fuzzy membership degree of each cluster in each set is depicted in Figure 2.3. As can be seen in Figure 2.2, in spite of the high degree of fuzzy intersection between clusters, an acceptable classification can be obtained with a fuzzy cluster if an efficient clustering method is applied.

Among all the variety of fields in which clustering algorithms can be applied, maybe one of the outstanding applications can be seen in industrial process control. Mostly a good control design for an industrial process needs a good estimation of the model of the plant. Clustering methods can play an important role in building good system modeling estimators such as fuzzy systems which are universal approximators of nonlinear systems. In real conditions for controlling a process, frequently only real input/output data are available, not a model of the process. Fuzzy systems have internal data to characterize the antecedent parameters of the best membership functions, as well as fuzzy rules, and may benefit from the definition of an optimal number of clusters in different parts of their structures. Therefore, the utilization of efficient robust clustering techniques for fuzzy applications can be an important issue.

This thesis is going to use T-S fuzzy modeling techniques in control design. As a requirement, finding an efficient clustering technique is considered. This thesis will

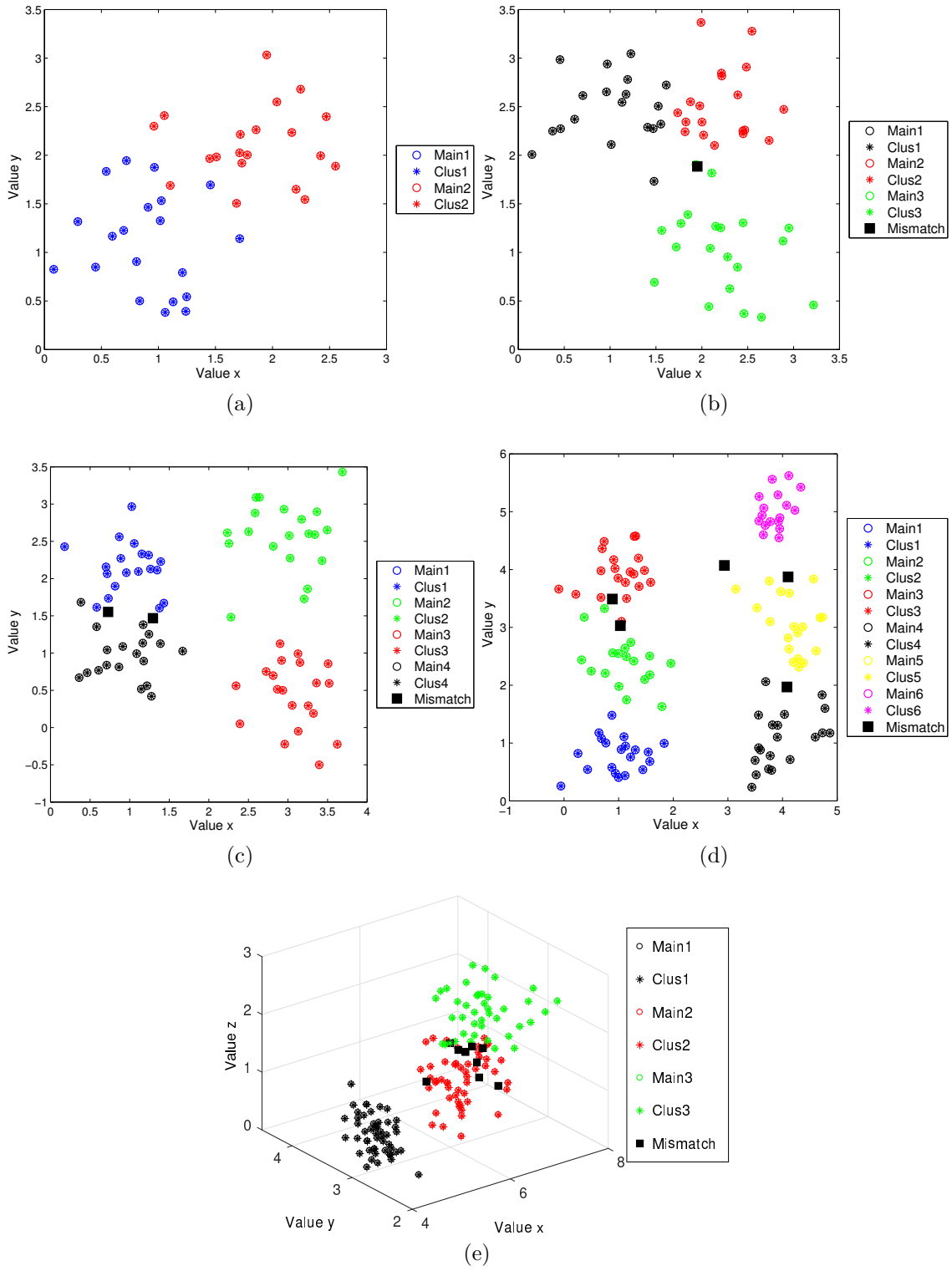


Figure 2.2: Fuzzy clustering: simulation results of the fuzzy clustering algorithm on the S_1-S_5 IRIS data sets: (a) S_1, (b) S_2, (c) S_3, (d) S_4, and (e) S_5. "Main" refers to clusters in the database, "Clus" refer to clusters obtained by the fuzzy cluster, and "Mismatch" correspond to wrongly clustered samples.

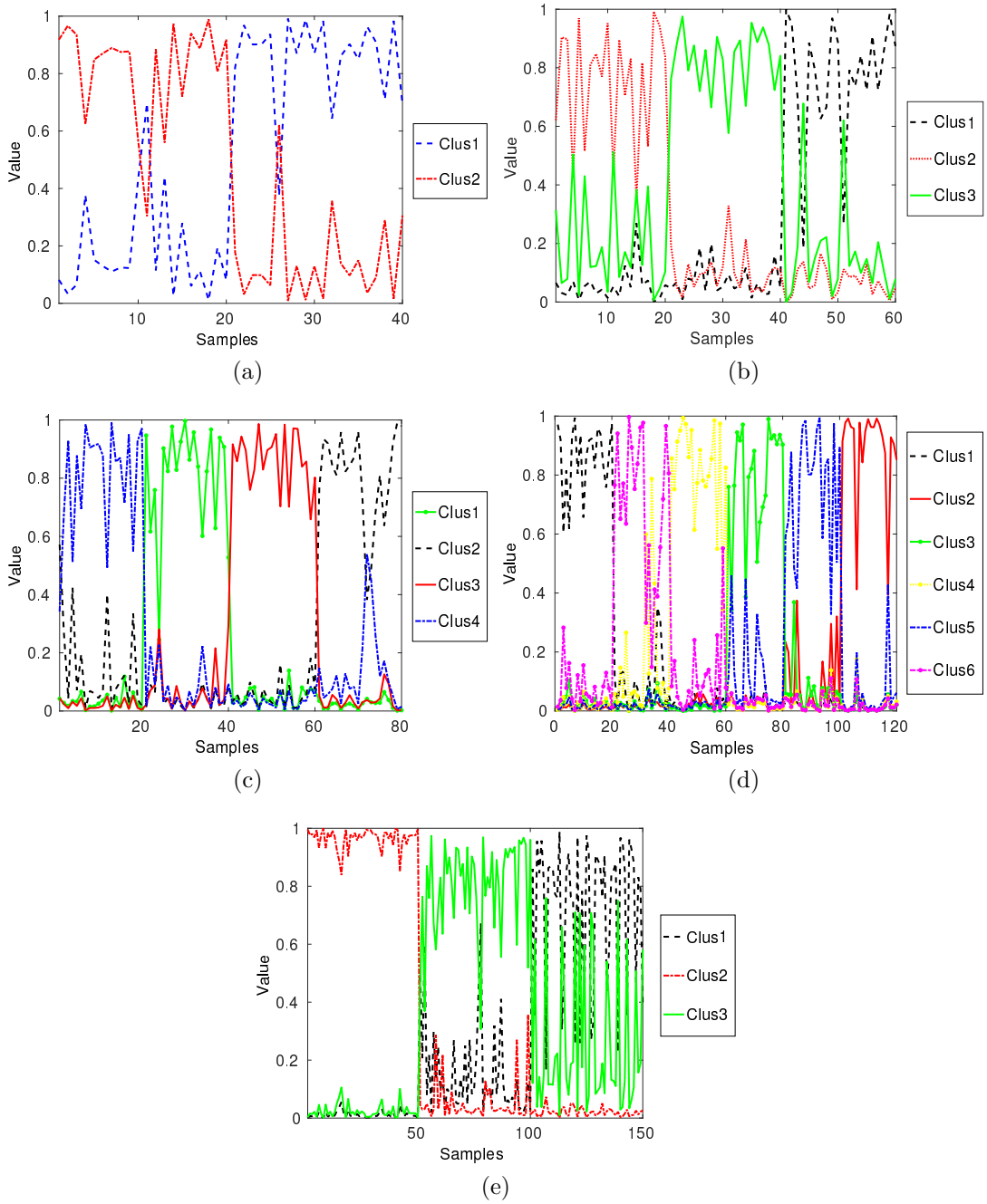


Figure 2.3: Fuzzy membership functions: simulation results of the fuzzy clustering algorithm on the S_1-S_5 IRIS data sets: (a) S_1, (b) S_2, (c) S_3, (d) S_4, and (e) S_5.

propose a hybrid unsupervised clustering method based on the K-nearest neighbor (KNN) and fuzzy C-means (FCM) methods. The number of clusters is selected by the algorithm. Two steps are involved for each candidate number of clusters. In the first step, the algorithm finds the initial clustering and respective centers by a KNN method. In the second step, the quality of the clustering is evaluated using the fuzzy clustering validation index with the goal of searching for the optimal clustering solution. The clustering methodologies proposed in this thesis can be involved in a wider variety of applications in a number of fields such as image segmentation, signal processing, motion detection, and robotics.

2.2 Fuzzy Systems

An overview of the main concepts of fuzzy systems is given in this section. However, for a better understanding and additional information on this topic, [Wang, 1997a] is recommended.

2.2.1 Concept

To construct fuzzy logic systems (FLS), expert's knowledge on the system needs to be expressed explicitly in the form of fuzzy IF-THEN rules. When the input to the fuzzy rules is given, the output is determined by inference using the fuzzy rules. FLS are characterized by a group of structures comprising four parts: knowledge-base, fuzzifier, inference engine, and defuzzifier, as can be seen in Figure 2.4. In the following subsections, a brief explanation of the four main parts of a FLS as it was represented in Figure 2.4 will be performed.

2.2.2 Knowledge-Base

In the *knowledge-base* "facts" are represented through linguistic variables and fuzzy logic rules. Once it is found that the knowledge of a specialist can be expressed through linguistic variables and rules of thumb, that involve imprecise antecedents and consequents, then the basis for a *knowledge-base* is established. The *knowledge-base* comprises the information that characterizes and defines the fuzzy system, and is composed of a data base, and a rule base containing a set of fuzzy rules.

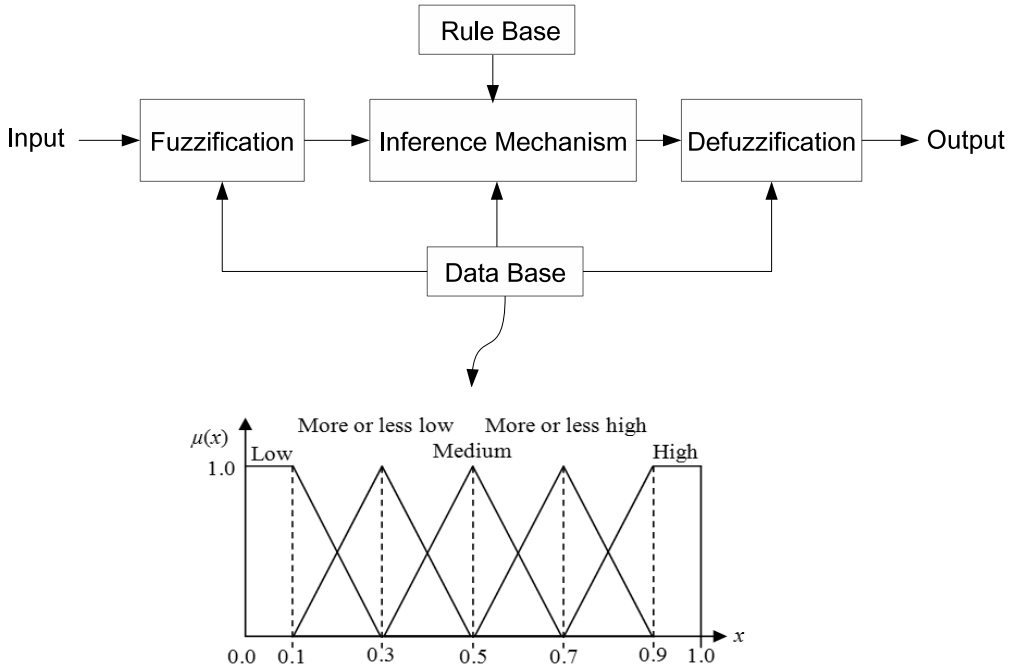


Figure 2.4: Basic configuration of a fuzzy logic system.

The fuzzy rules are composed of two parts: the antecedent (IF part) and the consequent (THEN part). For example, one could make up a rule that says: if temperature is high and humidity is high then room is hot. A knowledge-base composed of a set of N fuzzy IF-THEN rules R_i can be represented in the form:

$$R_i : \text{IF } x_1(k) \text{ is } A_1^i, \text{ and } \ldots \text{ and } x_n(k) \text{ is } A_n^i \text{ THEN } u(k) \text{ is } B_i, \qquad (2.1)$$
$$i = 1, \ldots, N,$$

where x_j $(j = 1, \ldots, n)$ are the fuzzy system input variables, and u is the output of the fuzzy system, A_j^i and B_i are the linguistic terms characterized by fuzzy membership functions $\mu_{A_j^i}(x) = U_j \rightarrow [0, 1]$ and $\mu_{B_i}(u) = V \rightarrow [0, 1]$, respectively, for $i = 1, \ldots, N$, and $j = 1, \ldots, n$. n is the number of input variables. $U_j \subset \mathbb{R}$ is the universe of discourse of x_j, for $j = 1, \ldots, n$, and $V \subset \mathbb{R}$ is the universe of discourse of u. The most commonly used membership function types are the trapezoidal, triangular, Gaussian, and generalized membership functions (MF), as represented in Figure 2.5. In this thesis only the Gaussian type of membership function is considered.

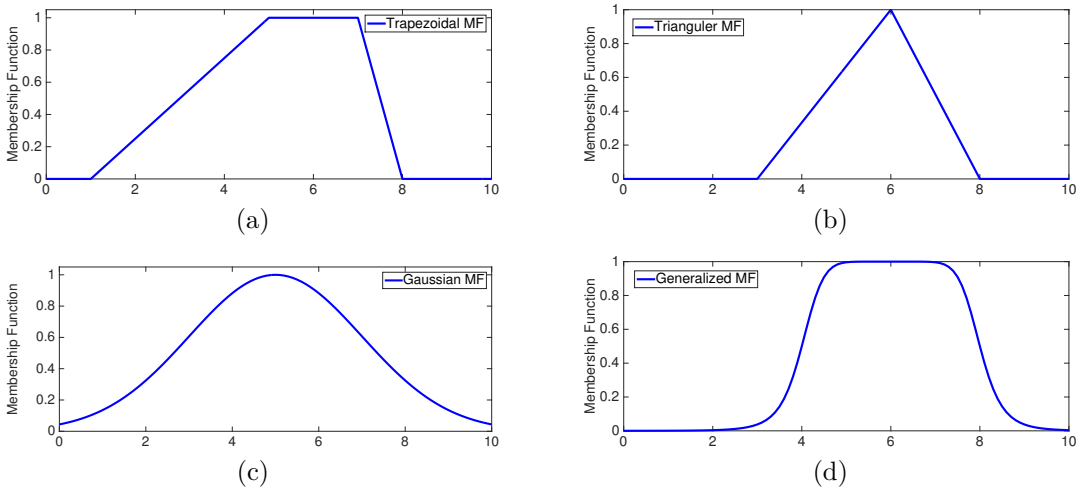


Figure 2.5: Examples of membership functions: (a) trapezoidal, (b) triangular, (c) Gaussian, and (d) generalized.

2.2.3 Fuzzifier

Another component of a fuzzy logic system is the *fuzzifier* which transforms input real numbers into fuzzy sets, taking the real-valued inputs and determining the degree to which they belong to each of the appropriate fuzzy sets via membership functions. In industrial systems, for example, the purpose of fuzzification is to map the real-valued inputs from a set of sensors or features of those sensors such as amplitude or spectrum, into values from 0 to 1 using a set of input membership functions.

The output of the *fuzzifier* unit reflects the degree to which each part of the antecedent is satisfied for each rule. In the present thesis, only the *singleton fuzzifier* (2.2) is considered, due to its simplicity of implementation. However, other fuzzifier methods can be consulted in [Wang, 1997a]. The mapping performed by the *singleton fuzzifier* is defined as follows:

$$\mu_{A'}(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} = \mathbf{x}^*, \\ 0, & \text{other cases.} \end{cases} \tag{2.2}$$

In (2.2), the fuzzifier transforms the real-valued input vector $\mathbf{x}^* \in S \subset \mathbb{R}^n$ into a fuzzy set A' defined in a universe of discourse S.

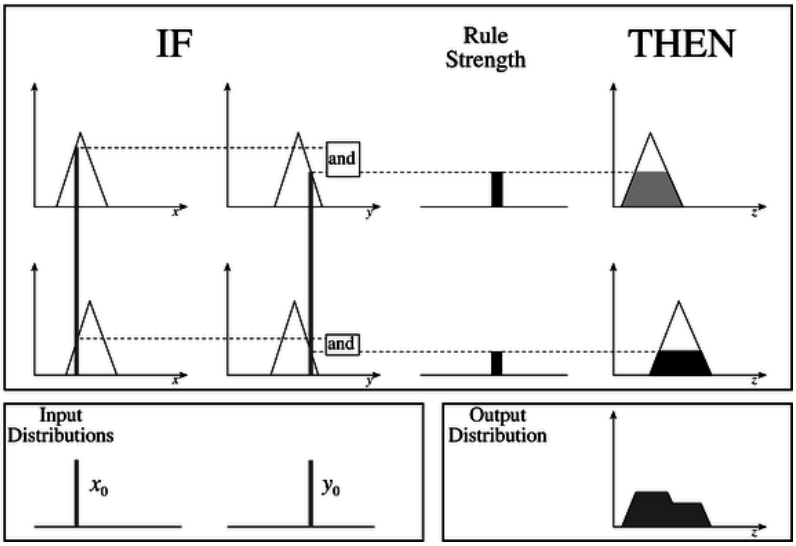


Figure 2.6: An example of a two input, and two rules Mamdani-minimum FIE with crisp inputs.

2.2.4 Fuzzy Inference Engine

The fuzzy inference engine (FIE) is one of the most important components of a fuzzy system. This section describes it in a simplified and compact manner. A more detailed description and formulation can be found in [Wang, 1997a].

If the antecedent of a given rule (2.1) has more than one part, fuzzy operators are applied to obtain one number that represents the result of the antecedent for that rule. In (2.1) only "and" operators are included in the antecedent. The input to the fuzzy operator is two or more membership values obtained from the fuzzified input variables, and the output is a single truth value. This number is then applied to the output membership function. In fuzzy logic, the basic fuzzy operators used to process the antecedent part of a rule can be the following: *intersection*, *union*, and *complement*. There is a variety of choices for the FIE, depending on the employed operators and on the employed implication and aggregation methods. For a more detailed description and formulation and a better understanding on FIEs and on different FIEs, [Wang, 1997a] is recommended.

A graphical example of a Mamdani-minimum FIE is given in Figure 2.6. This FIE contains two inputs, x_0 and y_0 shown at the lower left corner of the figure. This type of fuzzy system is not used in this thesis, but it is a good illustrative base example. The inputs are mapped into fuzzy numbers by drawing a line up from the

inputs to the input membership functions above and marking the intersection point. In this example, the "min" fuzzy "and" operator is used to combine the antecedent membership functions to compute the rule strength. Notice that the outputs of all the fuzzy rules must be combined to obtain one overall output fuzzy membership function. This is usually, but not always, done by using the fuzzy "or" operator. Figure 2.6 shows an example of this. The output membership functions on the right hand side of the figure are combined using the "max" fuzzy "or" operator to obtain the output distribution shown on the lower right corner of the figure.

2.2.5 Defuzzifier

In many cases, it is desired to obtain a single crisp output from a FIE unit. Fuzziness is involved in the rule evaluation during the intermediate steps, but the final desired output for each variable is generally a single number. However, a fuzzy set aggregate encompasses a range of output values, and so must be defuzzified in order to resolve into a single real-valued output to represent the output fuzzy set, and therefore the output of the fuzzy logic system. For example, suppose that one is trying to classify a letter drawn by hand on a drawing tablet, ultimately the FIE would have to come up with a crisp number to tell the computer which letter was drawn. This crisp number is obtained in this process called defuzzification.

There are several techniques to perform defuzzification. Some of the most common techniques to perform defuzzification are the centroid, mean of centers, bisector, middle of maximum (the average of the elements of the universe of discourse where the membership function attains the maximum), largest of maximum, and smallest of maximum. Perhaps the most popular defuzzification method is the centroid, which returns the centroid of the universe of discourse of the fuzzy set when considering the mass density to be defined by the fuzzy membership function.

Chapter 3

Overview of Adaptive Fuzzy Control and Robust Control Design

Contents

3.1 Fuzzy Based System Identification

Nonlinear system identification plays an important role in industrial processes. The application of an identified model of good quality can lead to a control design of higher quality. There are different nonlinear identification techniques. However, among them, relatively new methodologies based on fuzzy logic models were gradually becoming established not only in the academia but also in industrial appli-

cations. Fuzzy system identification is an example of a gray-box technique on the boundary between nonlinear black-box and qualitative models. Adequately designed fuzzy systems can simulate a nonlinear system with an acceptable accuracy. Powerful tools useful to contribute to the construction of fuzzy models from data sets can be found in the various fields of fuzzy logic, approximate reasoning, neural networks, pattern recognition, and regression analysis. Modern processes exhibit nonlinear behaviors, while most of them are exposed to complex characteristics such as time-varying dynamics, disturbances, constraints, and stochastic phenomena. Also, the multivariable nature of the systems can be seen frequently in real processes.

Several steps must be considered for having a successful system identification design [Nelles, 2000]. Some factors such as the choice of the model inputs, choice of the excitation signals, choice of the model structure, as well as the choice of the model parameters play significant roles to achieve an adequate system model by identification. The assumption that the performance of MPC methods is dependent on the availability of an accurate model may present problems, because many complex plants are difficult to be modeled mathematically based on physical laws, or have large uncertainties and strong nonlinearities. Having a successful system identification can be difficult to achieve, for example in complex industrial processes. Several types of approaches to modeling nonlinear plants can be considered to be used for MPCs. Among them, fuzzy models have received particular attention in the area of nonlinear modeling.

The principles of fuzzy sets and fuzzy logic were developed by Lotfi A. Zadeh in 1965 [Zadeh, 1965]. In the late 1960s and early 1970s, fuzzy set theory has grown to become a major scientific domain. Several new fuzzy logic methods such as fuzzy algorithms, fuzzy decision making, etc, were proposed, and grew as independent fields. In this period, Zadeh proposed fundamental concepts in fuzzy theory. Zadeh proposed the concepts of fuzzy algorithms in 1968 [Zadeh, 1968], fuzzy decision making in 1970 [Bellman and Zadeh, 1970], and fuzzy ordering in 1971 [Zadeh, 1971]. An important paper published by Zadeh was [Zadeh, 1973], where the author introduced the concepts of linguistic variables and proposed the use of fuzzy IF-THEN rules to formulate human knowledge, which established the foundation for fuzzy control. In the continuously increasing progress in fuzzy control systems, Mamdani and Assilian [Mamdani, 1974] developed the first fuzzy Logic control system [Mamdani and Assilian, 1975], to be used in a small steam engine. Until now many

fuzzy logic based models for system identification were proposed. Among them, one of the famous fuzzy models which has received interest from control engineers is the Takagi-Sugeno (T-S) type of fuzzy model [Takagi and Sugeno, 1985]. The Takagi-Sugeno fuzzy model can estimate complex nonlinear systems with few rules and accurately. The capacity of approximating complex nonlinear systems with few rules and high modeling accuracy, the possibility to apply conventional linear system theories for system analysis and synthesis, as well as the possibility to be used in both online and offline mode, all cause the T-S fuzzy model to become a popular nonlinear system estimator, being considered a relevant research area.

Many optimization methods were investigated to be used in T-S fuzzy modeling. Among them, evolutionary algorithms such as particle swarm optimization (PSO) algorithms and genetic algorithms (GAs) have shown a good adaptation to search for optimal T-S fuzzy model parameters [Delgado *et al.*, 2001; Coelho and Herrera, 2007; Shuzhi *et al.*, 2012]. Although a GA may be able to find the global minimum, it consumes too much search time. The PSO algorithm is an alternative. It can result in a lower computational complexity with a better performance of fitness evaluation in general when compared to GAs [Eberhart and Shi, 1998], consequently resulting in a better prediction performance when applied to learn a T-S fuzzy model.

3.1.1 Takagi-Sugeno Fuzzy Model Identification

Rule-based fuzzy models are categorized into two main types: Mamdani fuzzy models and Takagi–Sugeno (T-S) fuzzy models. The consequent parts of Mamdani fuzzy models are fuzzy sets. Differently, the consequent parts of the T-S fuzzy model are functions of the input variables. In the remaining of this thesis, linear consequent functions will be considered in T-S fuzzy systems. In many examples it was observed that comparing with Mamdani fuzzy models, T-S fuzzy models can approximate complex nonlinear systems with fewer rules and higher modeling accuracy. This advantage causes the T-S fuzzy model to become a popular nonlinear system estimator, being considered an active research area. Additionally, the Takagi-Sugeno fuzzy model combines the linguistic description with standard functional regression, while the antecedents describe fuzzy regions in the input space in which the consequent functions are valid. Consequent functions of the T-S fuzzy model are often based on a linear structure. Thus, conventional linear system theory can be used

for the system analysis and synthesis accordingly. This advantage gives another motivation which causes the T-S fuzzy models to be considered as a powerful engineering tool which can be applied in both system modeling and process control. The main task in the identification of processes based on T-S fuzzy models is to partition the input space into a number of fuzzy sub-spaces and obtain the antecedent and consequent parameters of the model.

The T-S fuzzy model can be constructed on online and/or offline mode. However, an online learning mode can have superiority because unlike offline mode, the online process can incrementally construct and improve the model, and adapt the model to changes in the process being modeled. Considering this fact, T-S fuzzy models are going to be used for control design purposes, and in particular they will be used in integration with MPC designs. In this context, the use of offline and/or online T-S fuzzy modeling combined with the concept of MPC can be considered as an efficient technique. With this motivation in mind, this PhD work will also investigate the combination of both MPC and online T-S fuzzy modeling techniques with the goal of achieving a fuzzy predictive controller that can be used for online process control in nonlinear industrial plants.

3.2 Hierarchical Fuzzy Modelling Structure

Application of fuzzy logic techniques for system identification and/or control of industrial processes are considered as efficient techniques due to the practical applications of fuzzy logic systems in identification and control, and to theoretical approximation properties of fuzzy systems. However, the design of a fuzzy model is a time-consuming and challenging process which involves expert knowledge acquisition, the fuzzy system structure definition, finding the rules, and fuzzy sets construction. From the past decades until now, the methodologies proposed to establish fuzzy logic systems have been suffering from a shortage in performance. Typically, in a standard fuzzy system, the number of rules increases exponentially with the number of variables [Lee *et al.*, 2003], and many research studies have been performed to address this important concern in the design of fuzzy logic systems. The initial objective was to try to find a way to reduce the total number of involved rules, and their corresponding computation requirements. If there are n input variables and m membership functions for each variable, then it needs m^n rules

to construct a fuzzy logic controller based on an exhautive partitioning of the input space based on all the combinations of input fuzzy sets for each variable. When n increases, the rule base will quickly overload the memory, and the computational time needed for implementing the fuzzy controller will increase significantly. In this context, the "curse of dimensionality" phenomenon arises, where the complexity of the problem increases exponentially with the number of variables involved. Additionally, as the number of antecedent fuzzy sets increases, the number of fuzzy rules increases polynomialy. Thus, in both the cases of increasing n or m the number of rules and the associated costs increase significantly, in a phenomenon called "rule explosion".

Some investigations have been performed to avoid rule explosion in fuzzy inference engines by converting an intersection rule configuration (IRC) to a union rule configuration (URC). For example, [Weinschenk *et al.*, 2003] presented a new mapping of classical fuzzy systems into a union rule configuration. Later, [Weinschenk *et al.*, 2004] provided a novel Fourier-based technique for designing a layered URC fuzzy system that eliminates the need for costly search techniques. As a practical efficient solution to overcome the problem of the "curse of dimensionality", and "avoidance of rule explosion", the idea of hierarchical fuzzy systems has been developed [Lee *et al.*, 2003]. Hierarchical fuzzy systems consist of a number of low-dimensional fuzzy systems in a hierarchical form. The hierarchical fuzzy systems have the advantage that the total number of rules increases only linearly with the number of input variables [Rajua *et al.*, 1991]. Later, the application of hierarchical fuzzy system concepts was improved when fuzzy systems with hierarchical structures were developed [Delgado *et al.*, 2009; Mendes *et al.*, 2012]. Despite having a successful performance, neither the works in [Delgado *et al.*, 2009] nor in [Mendes *et al.*, 2012] utilized any significant improvement in their interior structure to be augmented for online applications. Furthermore, for automatic extraction of all fuzzy logic system (FLS) parameters, the number of partitions of the input space is a key parameter. Approximately all the FLS's units of a T-S fuzzy model such as the set of fuzzy rules, the individual rules, the consequent parameters, as well as the unit of the inference mechanism, use this parameter during the FLS construction. However, in both the methods proposed in [Delgado *et al.*, 2009] and in [Mendes *et al.*, 2012], the number of partitions was defined firstly and then considered as a constant value during all the running time of the algorithm. With these motivations

in mind, this PhD work will investigate on a hierarchical T-S identification methodology which is designed in integration with hierarchical particle swarm optimization (HPSO) algorithm usable in real-time processes. The HPSO automatically extracts the structure and all FLS's parameters of a T-S fuzzy model by using input/output data in such way that it does not require any previous knowledge of the process model, and in particular no previous knowledge concerning the number of rules and the number of antecedent fuzzy sets of the T-S fuzzy model.

3.3 Adaptive Fuzzy Control

Many available works regarding the application of fuzzy system concepts in control design show the efficiency of fuzzy controllers for controlling industrial processes. However, a non-adaptive fuzzy control design can easily meet difficulties when the plants have unknown and/or time-varying models and parameters. In this case, specifying the fuzzy rule base to construct the controller for such plants is not an easy task. Consequently, the main control objectives can not be addressed correctly. As one solution, adaptive techniques can be used in control design structures. Adaptive fuzzy logic controllers try to focus on automatic on-line synthesis and tuning of fuzzy controller parameters and/or structure by learning and adapting to the dynamics of the plant. The adaptation law can help the fuzzy controller to adjust itself to changing environments [Wang, 1997a].

In general, adaptive fuzzy control designs can be categorized in two different types. Depending on how a fuzzy controller is constructed, different adaptive procedures can be investigated. For example, one typical type of fuzzy control design is direct fuzzy control (DFC) [Mendes *et al.*, 2014; Wang *et al.*, 2016]. In direct fuzzy control, the parameters of the controller are initially constructed from human control knowledge. In most cases in DFC design, a fuzzy logic system (FLS) is considered for control purposes. A data set composed of the input command signal (the output of the controller), and the reference signal and the corresponding plant output when the system is under control (the inputs of the controller) is provided. In the next step, this data set is applied to the fuzzy logic system (FLS) to initialize a training process. For this type of fuzzy control design, usually computational intelligence techniques are used to directly adjust/adapt the controller, and thus achieve a direct adaptive fuzzy control (DAFC) design. An intelligent system which

uses an iterative procedure is adjusted to reduce the output error between the plant and a desired reference.

As another type, a fuzzy control can be constructed based on an explicit fuzzy modeling of the plant. Normally, a system modeling can be formulated based on different types of mathematical modeling techniques in different spaces such as state space or transfer function space. Among many system modelling representation types, fuzzy system modeling has an efficient applicable modelling structure which has been frequently used in system identification methodologies [Nelles, 2000]. In indirect adaptive fuzzy control (IAFC), the parameters of the fuzzy model of the plant are initially constructed from some human knowledge about the unknown plant, and then iteratively adjusted to reduce the output error between the plant and an estimated model, while the current model parameters are used to indirectly adapt the controller [Rastegar *et al.*, 2016b].

For the first time, an adaptive fuzzy controller which was called the linguistic self-organizing controller was introduced by [Procyk and Mamdani, 1978]. Later, the fuzzy model reference learning controller was introduced in [Layne and Passino, 1993], and has shown to have a successful performance in simulation [Layne and Passino, 1993; Kwong *et al.*, 1994; Passino *et al.*, 1995; Layne and Passino, 1996; Kwong and Passino, 1996; Lennon and Passino, 1999], and in implementation studies [Moudgal *et al.*, 1995; Zumberge and Passino, 1998]. The most studied stable adaptive fuzzy control (AFC) schemes are based on feedback linearization [Wang, 1996, 1992; Spooner *et al.*, 2002], which was considered for the control of nonlinear plants. More recently, application of adaptive fuzzy control can be seen in industry applications. In [Liu *et al.*, 2013], the problems of stability and tracking control for a class of large-scale nonlinear systems with unmodeled dynamics were addressed by designing the decentralized adaptive fuzzy output feedback approach. For stability analysis the Lyapunov stability method was used. It was shown that all the signals in the closed-loop nonlinear system are bounded, and the system outputs track the reference signals to a small neighborhood of the origin by choosing the design parameters appropriately. In [Li *et al.*, 2014b], the problem of adaptive fuzzy output-feedback control was investigated for a class of output constrained uncertain nonlinear systems with input saturation and unmeasured states. To address the input and output constraints, a barrier Lyapunov function and an auxiliary design system were employed, respectively. The adaptive strategy in the control design

was founded on a back-stepping recursive technique. Unlike some existing control schemes for systems with input saturation, the developed controller does not require assumptions on the available states. Fuzzy logic systems were utilized to approximate the unknown nonlinear functions, and a fuzzy state observer was designed to estimate the unmeasured states. The control design issue in both the works [Liu *et al.*, 2013; Li *et al.*, 2014b] was limited to tracking of reference signals at the origin.

3.4 Receding Horizon Control Design

The interest in receding horizon control (RHC), which is also known as model predictive control (MPC), began to increase after the introduction of the Identification and Command (IDCOM) Method [Richalet *et al.*, 1978] and the Dynamic Matrix Control (DMC) [Cutler and Ramaker, 1980]. Since then, MPC has become a widely used technology in process control. For example, the MPC has become a standard technique for control of multivariable constrained chemical processes. Applications of this technology have made it a multimillion dollar industry. Nowadays, a new crude distillation unit in a refinery is not conceived with other control scheme except MPC. MPC fundamentally is a control strategy based on a predictive model of the process under control, which is used as a basis to find a future trajectory of the input manipulated (command) variable in such a way as to minimize a performance index that depends on the predicted future behavior of the plant. Li *et al.* [2012] first combined a T-S fuzzy model with a predictive control methodology to the control the nonlinear behavior of a boiler-turbine. In continuation, this work used a strategy based on GA to solve the problem of constraints in the control of the boiler-turbine system. The work [Maeder *et al.*, 2009] tried to suggest an offset-free model predictive control which was able to track an asymptotically a constant reference. For that, an additional disturbance state was employed to be combined with the main system to make an augmented system to obtain disturbance estimates. Li *et al.* [2012] presented an offset-free output predictive control approach for nonlinear processes without considering the presence of external disturbances. Izadi *et al.* [2011] investigated Moving Horizon Estimation (MHE) and the Unscented Kalman Filter (UKF) as two methods for nonlinear parameter estimation. A framework was then formulated for integrating MHE/UKF based fault estimator with MPC to form an active fault tolerant control system for systems with nonlinear constrained

dynamics.

Application of MPC has not been limited just on certain dynamical systems. MPC could widely show an effective role in the control of uncertain systems. For uncertain systems, the MPC based approaches led to LMI optimisation problems and solutions [Kothare *et al.*, 1996] or used ellipsoidal robustly invariant sets [Kouvaritakis *et al.*, 2000; Muñoz-Carpintero *et al.*, 2015], probabilistic approach [Calafiore and Fagiano, 2013], or were based on having an online construction of polytopic tubes [Evans *et al.*, 2012; Fleming *et al.*, 2015].

The designs in some of these methods were limited to control designs to track zero setpoints, or did not consider the existence of unexpected disturbances [Fleming *et al.*, 2013], or in some cases, the MPC control design was constructed for periodic step-type disturbances [Li *et al.*, 2014a]. But, there is a more important concern with MPC based robust control design. The presence of uncertainty in a linear system can cause computational difficulties. In particular in the case of polytopic systems, the number of vertices of the uncertainty set grows exponentially in the length of the prediction horizon [Evans *et al.*, 2012]. Attempts to find a solution for this problem can be seen in some recent works such as [Fleming *et al.*, 2015]. When facing external unexpected disturbances besides of uncertainty, the analysis and insurance of closed-loop feasibility, stability, and robustness of such MPC methodologies becomes a more challenging, and difficult issue to solve.

As mentioned throughout this thesis, the assumption of the knowledge of an accurate model in MPCs, presents problems because many complex plants are difficult to be mathematically modeled based on physical laws, or have large uncertainties and strong nonlinearities. Thus, to find an automatic control design which just uses input/output data for model construction, is an interesting and important issue. None of the methods of [Li *et al.*, 2012; Maeder *et al.*, 2009; Zhang *et al.*, 2009; Li *et al.*, 2012, 2014a; Han *et al.*, 2012] have a full automatic control structure. All these methods have the limitation of not being able to perform automatic selection of variables and delays of the model of the system: pre-selection is performed. The variable selection process is usually manual and not accompanied with the accurate selection of the right time delays, probably leading to low-accuracy results. A variable with the correct delay may contain more information about the output, than one which does not consider any delay or which considers an incorrect delay [Souza *et al.*, 2010].

The MPC methodologies are designed based on a mathematical model of the plant. Among several modeling approaches which have been applied, fuzzy logic systems have received particular attention in the area of nonlinear modeling. The use of fuzzy models together with the concept of model predictive control is a promising technique because both techniques can have an intuitive human interpretation, and can be explained in simple terms to industrial human operators. Additionally, fuzzy systems have general approximation properties in the modeling of functions and systems [Wang and Mendel, 1992; Kosko, 1994]. Among several fuzzy modeling approaches, Takagi-Sugeno fuzzy models have gained much popularity in MPC because their rule consequents are real-valued functions.

3.5 Constrained Robust Control Design

Often, in a system identification process due to complexity in plant behavior, just an uncertain model can be estimated. One of the typical class of uncertain systems/models is the class of discrete-time linear parameter varying (DT-LPV) plants [Pannocchia, 2004; Li *et al.*, 2014a]. In nonlinear processes, the theory of Linear Parameter-Varying (LPV) systems suggests an efficient modeling framework. In general, LPV systems can be presented as an extended version of the class of Linear Time-Invariant (LTI) systems. In LPV systems, the relations of signals are considered to be linear, but the parameters in the description of these relations are assumed to be time-varying functions. In a discrete-time setting, LPV systems are generally described in a state-space (SS) domain. Practical application of this framework is oriented by the fact that LPV control design is well developed, extending results of optimal and robust LTI control theory to nonlinear, time varying plants.

Although, uncertainty in industrial process control can introduce difficulty in a control design, it is not the unique challenging issue for consideration. Besides uncertainty, a control design problem can be exposed to unknown disturbances or stochastic phenomena. Furthermore, often some limitations/constraints can be seen in the performance of control parameters in real industrial environments. Input/output signals saturation, as well as limitation in the variation of system state variables, all can impose some constraints which must be considered in control design for such complex plants. That is why the control of DT-LPV plants in the presence of constraints while the plants are affected by unknown disturbances, is a very active and

relevant research area.

MPC designs have been considering these concerns slightly [Chakrabarty *et al.*, 2013; Kirubakaran *et al.*, 2014]. However, in real control processes, the issue of the reduction of computational complexity due to objective functions and terminal constraints in MPC can emerge as a weakness of such control approaches. For example, giving a solution for an infinite-horizon linear quadratic programming problem with terminal constraints, which is a typical optimization problem in MPC methodologies, is a more difficult task when compared to a system that is not exposed to constraints. Several works have been performed with the goal of reducing the computational time of the MPC optimization algorithm for both offline and online modes [Bemporad *et al.*, 2002]. However, due to the complexity of the works, this research field is still considered as a challenging area for investigation [Qi *et al.*, 2015]. This was a reason to start to investigate on the possibility of combining these two control techniques, MPC and sliding mode control (SMC), to take advantage of the MPC capability of coping with system constraints and the SMC robustness properties [Wang *et al.*, 2013]. However, other common and important drawback of the SMC scheme, or the model predictive SMC design, is the existence of the chattering phenomena caused by the high frequency switching control law which needs to be seriously taken into account [Lee *et al.*, 2009; Feng *et al.*, 2014].

Frequently, for having a chattering-free sliding mode control, fuzzy logic techniques were applied [Roopaei and Zolghadri, 2009]. As another solution a saturation function is applied instead of a sign function [Lo and Chen, 1995]. Higher-order SMC also is another possible solution for having a chattering-free sliding mode control [Feng *et al.*, 2014]. Each of the aforementioned solutions has its own complexity. For methods based on fuzzy systems, to find fuzzy rules, select best input variables, as well as the calculation of fuzzy membership parameters are challenging tasks. Solutions based on higher-order models increase the computational costs significantly which is not desirable. For saturation functions based solutions, finding thresholds of saturation functions is considered another challenging issue. In most cases, it can be seen that fuzzy logic techniques were applied to find the optimal value of the threshold in saturation functions which increases the difficulty in control design. Totally, due to complexity of the design, the concern regarding chattering is still an important issue that should be considered and researched.

From the above discussion, it can be concluded that the study on robust control

designs for LPV systems, while the systems, besides uncertainty, are also exposed to unknown external disturbances, and system constrains is an important research area [Rastegar *et al.*, 2017c]. Investigation on this area opens a new gate towards another interesting research case study such as robustly positively invariant set theory applications to find an appropriate robust control region.

After thorough investigation, the author found that to have an efficient robust control design for LPV systems in the presence of unknown external disturbance sources and system constrains, synergetic control theory (SCT) can be considered as a powerful mathematical tool to solve this problem. Synergetic control theory (SCT) offers a control framework based on a theory conceived to control non-linear dynamical processes [Kolesnikov, 2014], [Kolesnikov, 1994, cited in [Kolesnikov, 2014]]. The SCT based controllers are coordinated with the internal characteristics of the systems. The synergetic control structure is designed to make the closed-loop control system converge to regions/sets of attraction that correspond to the control purposes, and then force the trajectories to stay on those regions. The author believes the resulting control design for LPV systems [Rastegar *et al.*, 2017c] can also be extended and applied for the control of nonlinear state dependent discrete time varying (NSDDTV) systems [Rastegar *et al.*, 2017a] and linear time varying (LTV) systems [Rastegar *et al.*, 2017d].

3.6 Summary/Conclusion

In this chapter a review of fuzzy system identification, and fuzzy control and their derivations for industrial applications was considered. Taking into account that one of the objectives of the thesis is the control of nonlinear industrial processes, the author considers that the use of fuzzy logic, hierarchical PSO algorithms, and model predictive control can result in efficient control techniques. MPC is a good control paradigm to apply in industrial processes. This controller has the capability to be combined with Takagi-Sugeno fuzzy models, and lends itself well to be used in multivariable control. Furthermore, typical systems restrictions such as state and control command constraints can be incorporated into the control methodology. Moreover, MPC can be integrated with an adaptive law to make an adaptive model predictive control design. Among many fuzzy logic system types, T-S fuzzy model is chosen in this thesis to be combined with MPC. The T-S fuzzy model can be identified by

hierarchical PSO algorithms yielding good results. Another main objective in this thesis is trying to have a constrained robust control design for uncertain time-varying LPV systems. For that purpose, the synergetic control theory is a strong candidate to be used in the design of a robust controller for LPV systems. The author believes that the use of synergetic control theory combined with positively invariant set theory can lead to a robust control framework, in such a way that the total proposed control methodology can deal with system uncertainties, external disturbances, and system constraints. Moreover, the author believes that the resulting control design is also able to be extended to be used in the control of nonlinear state dependent discrete time varying (NSDDTV) systems and linear time varying (LTV) systems.

Chapter 4

Takagi-Sugeno Fuzzy Model Identification

Contents

4.1 Introduction / State of the Art

System identification based on experimental data has been considered a powerful practical engineering tool in many industrial control fields. A common point among most control methodologies is their assumption of the knowledge of an accurate model of the process to be controlled. This assumption may cause important problems, because many complex plants are difficult to be mathematically modeled based on physical laws or have large uncertainties and strong nonlinearities. Several types of approaches to modelling nonlinear plants can be considered. Among them, fuzzy models have received particular attention in the area of nonlinear modelling [Škrjanc, 2009], especially the Takagi-Sugeno (T-S) fuzzy models [Takagi and Sugeno, 1985]. The main feature of a T-S fuzzy model is that it may express the local dynamics on the region of the tate-space associated to each fuzzy implication (rule) by a linear system model [Barros and Dexter, 2007; Chang *et al.*, 2010; Hua *et al.*, 2013; Dovzan *et al.*, 2015]. The T-S fuzzy model parameters can be estimated in either or both offline or online modes. However, an online learning mode can have superiority because in most cases, the collected data set used in offline methods is limited, and the estimated T-S fuzzy model may not provide adequate accuracy in parts of the, or the whole, operating areas of the plant [Salahshoor *et al.*, 2012; Li and Du, 2012]. Many optimization methods were investigated to be used in T-S fuzzy modelling. Among them, evolutionary algorithms such as particle swarm optimization (PSO) algorithms and genetic algorithms (GAs) have shown a good adaptation to search for optimal T-S fuzzy model parameters [Delgado *et al.*, 2001; Coelho and Herrera, 2007; Shuzhi *et al.*, 2012]. Although a GA may be able to find the global minimum, it consumes too much search time. The PSO algorithm is an alternative. It can result in a lower computational complexity with a better performance

of fitness evaluation in general when compared with GAs [Eberhart and Shi, 1998], consequently resulting in a better prediction performance when applied to learn a T-S fuzzy model. Ali and Ramaswamy [2009] presented an optimal fuzzy logic control (FLC) algorithm for vibration mitigation of buildings using magneto-rheological (MR) dampers. A micro-genetic algorithm (m-GA) and a standard particle swarm optimization (SPSO) were used to optimize the FLC parameters. In the method, only the membership function parameters are optimized, while the other components of the fuzzy system are considered to be fixed. An adaptive PID controller based on adaptive PSO was proposed in [Alfi and Modares, 2011]. In most cases, controlling the inertia weight of PSO has been done based on a constant parameter, or a linearly decreasing inertia weight PSO (LDW-PSO) has also been used [Shi and Eberhart, 1998]. But in [Alfi and Modares, 2011], the inertia weight was dynamically adapted for every particle by considering a measure called the adjacency index (AI). This idea lead the classical LDW-PSO algorithm to be evolved into a new class of system modelling based on an adaptive PSO (APSO) methodology. The results proved a superiority of APSO performance when compared to the LDW-PSO or GA.

The prediction performance of a Takagi-Sugeno fuzzy model depends on its complexity (e.g. number of fuzzy rules), on the number and type of membership functions, on the antecedent variables, and on the consequent regressors [Yao *et al.*, 2014]. A hierarchical genetic algorithm (HGA) was proposed by Delgado *et al.* [2001] to find optimal parameters of T-S fuzzy systems through an evolutionary genetic algorithm and a neuro-based technique, and then was improved [Delgado *et al.*, 2009] by proposing an idea for pre-selection of input variables using an auxiliary criterion. However, the variable and delay selection are not jointly performed with the learning of the fuzzy model, which precludes the global optimization of the prediction setting. As an evolution of the work [Delgado *et al.*, 2001, 2009], Mendes *et al.* [2012] proposed an identification methodology based on the application of a HGA, and also the proposed identification method was applied for the design of a control methodology [Mendes *et al.*, 2014]. The main advances in this work are the improvement of the whole hierarchical structure, automatically extracting the fuzzy control rules. Despite of having a successful performance, the HGA did not utilize any significant improvement in its interior structure to be augmented for online applications. A related methodology for online applications was proposed in [Mendes *et al.*, 2013], using a combination with other method, the RLS method with adaptive

directional forgetting (ADF) [Kulhavý, 1987]. For automatic extraction of all FLS parameters, the number of fuzzy partitions is a key parameter. Approximately all the FLS units of a T-S fuzzy model such as the set of fuzzy rules, the individual rules, the consequent parameters, as well as the unit of the FLS, use this parameter during the FLS construction. However, in the methods proposed in [Delgado *et al.*, 2001], [Mendes *et al.*, 2012], [Mendes *et al.*, 2013], and [Mendes *et al.*, 2014], the number of partitions is defined initially, and then considered as a constant value during all the operation of the algorithm. Abbasi *et al.* [2015] represented a time delay fuzzy Takagi-Sugeno (T-S) for modeling and analyzing the human immunodeficiency virus type 1 (HIV-1) dynamics. In [Abdelmalek *et al.*, 2016], a strategy for the detection and isolation of current sensor faults in the stator of a double-fed induction generator (DFIG) based Wind Turbine (WT) was implemented. The work develops local linear models of a DFIG by applying T-S fuzzy modeling in the operating region of the wind turbine. The resulting T-S fuzzy DFIG model was used with a bank of Luenberger observers to detect current sensor faults in the stator. Large nonlinearities can be seen in the dynamics of both the HIV-1 process model [Abbasi *et al.*, 2015] and the DFIG [Abdelmalek *et al.*, 2016]. However, the constructed T-S fuzzy model in both cases does not utilize any adaptive procedure in these works.

4.2 T-S Fuzzy Model Identification Methodology

This section proposes an algorithm to identify a T-S fuzzy model to approximate unknown nonlinear processes based on input/output data. The proposed method identifies the structure and parameters of the model. The fuzzy rules, the parameters of the antecedent fuzzy membership functions, and the consequent parameters are automatically learned from system data. The proposed hybrid identification methodology can be considered for application in problems such as the design of data-driven soft sensors, or in model-based predictive control.

4.2.1 Modelling Using T-S Fuzzy Models

Takagi-Sugeno fuzzy models with simplified linear rule consequents are universal approximators capable of approximating any continuous nonlinear system [Ying,

1997]. For more details about T-S fuzzy models, references [Takagi and Sugeno, 1985; Wang, 1997b], are recommended. With a T-S fuzzy model, the global operation of the nonlinear system can be accurately approximated into several local affine models. In general, a nonlinear system can be described by a T-S fuzzy system model defined by the following fuzzy rules:

$$\begin{aligned} R_i: \text{ IF } & x_1(k) \text{ is } A_1^i, \text{ and } \ldots \text{ and } x_N(k) \text{ is } A_N^i \\ & \text{THEN } y_i(k) = \theta_{i1}x_1(k) + \cdots + \theta_{iN}x_N(k), \quad i = 1, 2, \ldots, c, \end{aligned} \tag{4.1}$$

where R_i represents the i-th fuzzy rule ($i = 1, 2, \ldots, c$), c is the number of rules, $x_1(k), \ldots, x_N(k)$ are the input variables of the T-S fuzzy system. A_j^i ($j = 1, 2, \ldots, N$) are linguistic terms characterized by fuzzy membership functions $\mu_{A_j^i}(x_j)$ which describe the local operating regions of the plant. $\theta_{i1}, \cdots, \theta_{iN}$ are model parameters of $y_i(k)$. From (4.1), the model output $y(k)$ can be written as

$$\begin{aligned} y(k) &= \sum_{i=1}^{c} \bar{\omega}^i[\mathbf{x}(k)]\mathbf{x}(k)\boldsymbol{\theta}_i, \\ &= \boldsymbol{\Psi}(k)\boldsymbol{\Theta}, \end{aligned} \tag{4.2}$$

where for $i = 1, \ldots, c$, and assuming Gaussian membership functions,

$$\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)], \tag{4.3}$$

$$\mu_{A_j^i}(x_j) = \exp\left(-\frac{(x_j - v_{ij})^2}{\sigma_{ij}}\right), \; j = 1, \ldots, N, \tag{4.4}$$

$$\bar{\omega}^i[\mathbf{x}(k)] = \frac{\prod_{j=1}^{N} \mu_{A_j^i}(x_j)}{\sum_{i=1}^{c} \prod_{j=1}^{N} \mu_{A_j^i}(x_j)}, \tag{4.5}$$

$$\boldsymbol{\theta}_i = [\theta_{i1} \ldots, \theta_{iN}]^T, \tag{4.6}$$

$$\boldsymbol{\Theta} = \left[\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_2^T, \ldots, \boldsymbol{\theta}_c^T\right]^T, \tag{4.7}$$

$$\boldsymbol{\Psi}(k) = \left[\left(\bar{\omega}^1[\mathbf{x}(k)]\right)\mathbf{x}(k), \ldots, \left(\bar{\omega}^c[\mathbf{x}(k)]\right)\mathbf{x}(k)\right], \tag{4.8}$$

where v_{ij} and σ_{ij} are the antecedent parameters, which represent the center and width of the antecedent membership functions, respectively, and which need to be defined/learned. Parameters v_{ij} and σ_{ij} will be learned from data using the Fuzzy C-means method presented in Subsection 4.2.2 below. The consequent parameters

θ_{ij} are estimated by a RLS method, with ADF (Subsection 4.2.5).

4.2.2 Fuzzy C-Means

The objective of the fuzzy c-means (FCM) clustering algorithm is the partitioning of a dataset $\mathbf{X}$ into a predefined number of clusters, c. In the fuzzy clustering methods, the objects can belong to multiple clusters, with different degrees of membership. Consider N samples which compose an observation l of the input variables (one sample of each input variable), which are grouped in an N-dimensional observation/sample vector $\mathbf{x}_l = \mathbf{x}(l) = [x_{l1}, \ldots, x_{lN}]^T \in \mathbb{R}^N$. A set of L observations/objects is then denoted as

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1N} \\ x_{21} & x_{22} & \ldots & x_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ x_{L1} & x_{L2} & \ldots & x_{LN} \end{bmatrix}. \tag{4.9}$$

For the T-S fuzzy model identification problem, each complete input/output observation $(\mathbf{x}_l, y(l)) = (\mathbf{x}(l), y(l))$, is composed of one input observation $\mathbf{x}(l)$, and one output observation $y_l = y(l)$, for $l = 1, \ldots, L$. Sections 4.2.2-4.2.4 deal with the clustering issue, and will use only the input part, $\mathbf{X}$, of the observations.

A fuzzy partition of the set $\mathbf{X} \in \mathbb{R}^{L \times N}$ into c clusters, is a family of fuzzy subsets $\{A^i \mid i = 1, \ldots, c\}$. The membership functions of these fuzzy subsets are defined as $\mu_i(l) = \mu_{A^i}(\mathbf{x}_l)$, and form the fuzzy partition matrix $\mathbf{U} = [u_{il}] = [\mu_i(l)] \in \mathbb{R}^{c \times L}$. The i-th row of the matrix $\mathbf{U}$ contains the values of the membership function of the i-th fuzzy subset A^i for all the observations belonging to the data matrix $\mathbf{X}$. The partition matrix has to meet the following conditions [Dovžan and Škrjanc, 2011; Mendes *et al.*, 2013]: The membership degrees are real numbers in the interval $\mu_i(l) \in [0, 1]$, for $l = 1, \ldots, L$; The total membership of each sample in all the clusters must be equal to one $\sum_{i=1}^{c} \mu_i(l) = 1$; And none of the fuzzy clusters is empty, neither do any contain all the data $0 < \sum_{l=1}^{L} \mu_i(l) < L$, for $i = 1, \ldots, c$. FCM clustering tries to minimize the following objective function, which has a pre-defined number of clusters, c, and includes a fuzziness weighting parameter, η:

$$J(\mathbf{X}, \mathbf{U}, \mathbf{V}) = \sum_{i=1}^{c} \sum_{l=1}^{L} [\mu_i(l)]^{\eta} d_{il}^2(\mathbf{x}_l, \mathbf{v}_i), \tag{4.10}$$

where $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_c]^T \in \mathbb{R}^{c \times N}$ is a matrix of cluster centroid vectors $\mathbf{v}_i = [v_{i1}, \cdots, v_{iN}]^T$, $d_{il}(\mathbf{x}_l, \mathbf{v}_i)$ is the Euclidean distance (l^2-norm) between the observation $\mathbf{x}_l$ and the cluster centroid $\mathbf{v}_i$, and the overlapping factor, or fuzziness parameter, η, influences the fuzziness of the resulting partition. The partition can range from a hard partition ($\eta = 1$) to a completely fuzzy partition ($\eta \to \infty$). In order to find the fuzzy clusters in the dataset $\mathbf{X}$, equation (4.10) must be minimized. If the derivative of the objective function is taken with respect to the cluster centers $\mathbf{V}$ and to the membership values $\mathbf{U}$, then optimum membership values are calculated as follows [Dovžan and Škrjanc, 2011]:

$$\mu_i(l) = \left(d_{il}^2 \sum_{q=1}^{c} \left(d_{ql}^2 \right)^{1/(\eta-1)} \right)^{-1}, \tag{4.11}$$

where

$$d_{il}^2 = (\mathbf{x}_l - \mathbf{v}_i)^T (\mathbf{x}_l - \mathbf{v}_i), \tag{4.12}$$

and

$$\mathbf{v}_i = \frac{\sum_{l=1}^{L} \mu_i^{\eta}(l) \mathbf{x}_l}{\sum_{l=1}^{L} \mu_i^{\eta}(l)}. \tag{4.13}$$

The v_{ij} parameters of (4.4) are obtained from the center-vectors $\mathbf{v}_i = [v_{i1}, \cdots, v_{iN}]^T$ of (4.13). To finalize the identification of the premise parameters in (4.4) of the T-S model (4.1)-(4.2), the $\boldsymbol{\sigma}_i = [\sigma_{i1}, \cdots, \sigma_{iN}]^T$, $i = 1, \ldots, c$, can be calculated from $\mathbf{U} = [\mu_i(l)]$, as follows:

$$\sigma_{ij} = \sqrt{\frac{2 \sum\limits_{l=1}^{L} \mu_i(l)(x_{lj} - v_{ij})^2}{\sum\limits_{l=1}^{L} \mu_i(l)}}, \quad j = 1, \ldots, N. \tag{4.14}$$

4.2.3 Fuzzy Validity Indices

Since FCM is an unsupervised clustering algorithm, some cluster validity index is required to evaluate the quality of the clustering that results from the algorithm. Each index is categorized based on specified criteria. Several types of cluster validity criteria exist, namely external criteria which are based on the external characteristics regarding all of the data, internal criteria which use quantities internal to data of each class, and relative criteria which basically follow the idea of evaluating a clus-

tering algorithm by comparing it to other clustering schemes. However, all of these criteria can be converted into two main optimal clustering criteria, compactness and separation [Berry and Linoff, 1996].

The compactness criteria have been proposed based on the idea that the members in one same cluster should be as close to each other as possible. For the compactness criteria the variance, which should be minimized, is one common measure. Some conventional fuzzy validity indices for this class of scheme are for example the partition coefficient (PC) [Bezdek, 1974], the partition entropy [Bezdek, 1975], and the proportion exponent [Windham, 1982]. The main difference between these three fuzzy validity indices are in the way that the membership functions of fuzzy sets are combined and used for fuzzy evaluation. For example, while the Partition coefficient (PC) uses the square of membership functions of fuzzy sets in the formulation, the partition entropy validity indice uses logarithmic type of membership functions of fuzzy sets. Also, in the formulation of fuzzy evaluation of the proportion exponent, there is an exponential-type distance term which causes more robust clustering results.

Separation criteria are organized based on the distance between the closest members of the clusters, the distance between the most distant members, or the distance between the centers of the clusters. A conventional performance index in this class of scheme is the Xie-Beni index [Xie and Beni, 1991]. Although the Xie-Beni index has proved that it can provide a reliable response over a wide range of choices for the number of clusters and the fuzziness weighting parameter, the Xie-Beni index has two intrinsic drawbacks: 1) the validation index monotonically decreases when the number of clusters gets very large and close to the number of data points, and 2) strong interaction between the cluster validity index and fuzziness weighting parameter η imposes unpredictable behaviour in the results when the fuzziness weighting parameter approaches infinity. The first problem was considered by Kwon [1998], who imposed a punishing function to eliminate the decreasing tendency. Tang and Sun [2005] proposed an improved validation index for the FCM algorithm to overcome the above two problems with the same idea. This work uses the Tang-Sun's

validation index which is defined as follows:

$$V_T(\mathbf{U},\mathbf{V};\mathbf{X}) = \frac{\sum_{i=1}^{c}\sum_{l=1}^{L}\mu_{il}^2 \parallel \mathbf{x}_l - \mathbf{v}_i \parallel^2}{\min\limits_{i\neq k} \parallel \mathbf{v}_i - \mathbf{v}_k \parallel^2 + 1/c} + \frac{\frac{1}{c(c-1)}\sum_{i=1}^{c}\sum_{k=1,k\neq i}^{c} \parallel \mathbf{v}_i - \mathbf{v}_k \parallel^2}{\min\limits_{i\neq k} \parallel \mathbf{v}_i - \mathbf{v}_k \parallel^2 + 1/c}, \tag{4.15}$$

where c is number of clusters, and $\mathbf{v}_i$ is the center of cluster i, $i = 1, ...c$. The numerator of the second term in (4.15) is an *ad hoc* punishing function (average distance between cluster centers) which is applied to eliminate the decreasing tendency of $V_T(\cdot)$ as $c \to L$.

4.2.4 K-Nearest Neighbor (K-NN)

The FCM algorithm with pre-defined initial values such as the number of clusters, initial cluster centers and fuzziness weighting exponent η converges to a solution at which the objective function J in (4.10) is minimized. In practice, in many cases, randomly choosing initial FCM parameters may cause the FCM to just obtain results which are only locally optimal [Yu *et al.*, 2004; Wu, 2012]. To overcome these problems, an initialization technique based on a K-NN method is proposed to initialize the FCM method [Algorithm 4.1, Steps 1-3(c)]. The basic FCM follows an iterative procedure to converge to a solution. But with the proposed initialization procedure just one iteration of the FCM is enough to learn the antecedent parameters in the course of learning the T-S fuzzy model [Algorithm 4.1, Step 3(d)]. The basic K-NN is a non parametric learning algorithm for classification some initialization of the 'K' decides how many neighbours influence the classification. In the initialization procedure proposed in this work, the K is a parameter to be determined which will be iteratively calculated by the proposed method in order to obtain the best/optimal value, c^*, for the number of clusters c subject to a maximum value c_{max} (see Algorithm 4.1, Steps 3). Thus, the resulting K for the best c is obtained in Algorithm 4.1. The complete proposed method is explained in Subsection 4.2.6. In Subsection 4.2.6 it will be explained how the combination of these steps together can result in the NUFCA to initialize the T-S fuzzy model.

4.2.5 Recursive Least Squares Method With Adaptive Directional Forgetting

In the proposed nonlinear systems modeling methodology, after learning the antecedent parameters, the consequent parameters are given by a recursive least squares (RLS) method, with the adaptive directional forgetting approach (ADF) (RLS-ADF) [Kulhavý, 1987; Bobál *et al.*, 2005; Mendes *et al.*, 2013] here adapted for the T-S fuzzy model. Using off-line training algorithms, the T-S fuzzy model can be obtained from input-output data collected from a plant. However, such collected dataset(s) can be limited, the obtained T-S fuzzy models maybe do not provide adequate accuracy, the system can be nonlinear and/or time-varying, or can have varying operating points and varying model parameters. Adaptive methodologies should be applied to solve these problems. The goals of having an adaptive forgetting factor is to improve convergence rate, tracking, and stability of RLS, as well as avoiding covariance matrix wind-up in case of poor excitation of the system.

At each iteration l, the vector of parameter estimates (4.6), is updated using

$$\boldsymbol{\theta}_i(l) = \boldsymbol{\theta}_i(l-1) + \frac{\mathbf{C}_i(l-1)\boldsymbol{\psi}_i^T(l)}{1+\xi_i}\left[y_i(l) - \boldsymbol{\psi}_i(l)\boldsymbol{\theta}_i(l-1)\right], \tag{4.16}$$

where $\boldsymbol{\psi}_i(l) = (\bar{\omega}^i[\mathbf{x}(l)])\,\mathbf{x}(l)$, $\xi_i = \boldsymbol{\psi}_i(l)\mathbf{C}_i(l-1)\boldsymbol{\psi}_i^T(l)$, $\mathbf{C}_i(l)$ is the covariance matrix of fuzzy rule i, and $y_i(l) = (\bar{\omega}^i[\mathbf{x}(l)])\,y(l)$. The initial value $\mathbf{C}_i(0)$, of $\mathbf{C}_i(l)$, should be set to a diagonal matrix where the main diagonal entries are suitably large numbers, as for example 10^5 for all main diagonal entries.

The covariance matrix is also updated at each iteration, l, using

$$\mathbf{C}_i(l) = \mathbf{C}_i(l-1) - \frac{\mathbf{C}_i(l-1)\boldsymbol{\psi}_i^T(l)\boldsymbol{\psi}_i(l)\mathbf{C}_i(l-1)}{\varepsilon_i^{-1}+\xi_i}, \tag{4.17}$$

where $\varepsilon_i = \varphi_i(l-1) - \frac{1-\varphi_i(l-1)}{\xi_i}$, and $\varphi_i(l-1)$ is the forgetting factor at iteration $(l-1)$ of the fuzzy rule i.

The adaptation performed on the forgetting factor is obtained using [Bobál *et al.*, 2005; Bobál and Chalupa, 2008; Kulhavý, 1987], [Kulhavý, 1985, cited in [Bobál and

Chalupa, 2008]]:

$$\varphi_i(l) = \frac{1}{1 + (1+\rho)\left\{\ln(1+\xi_i) + \left[\frac{(\nu_i(l)+1)\gamma_i}{1+\xi_i+\gamma_i} - 1\right]\frac{\xi_i}{1+\xi_i}\right\}}, \tag{4.18}$$

where $\nu_i(l) = \varphi_i(l-1)(\nu_i(l-1)+1)$, $\gamma_i = \frac{(y_i(l)-\boldsymbol{\psi}_i(l)\boldsymbol{\theta}_i(l-1))^2}{\tau_i(l)}$, $\tau_i(l) = \varphi_i(l-1)\left[\tau_i(l-1) + \frac{(y_i(l)-\boldsymbol{\psi}_i(l)\boldsymbol{\theta}_i(l-1))^2}{1+\xi_i}\right]$, and ρ is a positive constant. The initial values of $\varphi_i(0)$, $\tau_i(0)$ and $\nu_i(0)$ should be set between zero and one.

4.2.6 Proposed T-S Fuzzy Model Identification Based on an Unsupervised Fuzzy Clustering Algorithm

In this section a new T-S fuzzy model identification algorithm is proposed [Rastegar *et al.*, 2016b]. To construct a T-S fuzzy system of the form (4.2) it is necessary to obtain the number of rules, the antecedent membership functions, the set of rules, and also to learn and update the consequent parameters ($\boldsymbol{\theta}_i$). The antecedent part is given by a new unsupervised fuzzy clustering algorithm (NUFCA) and the consequent parameters are estimated by the RLS-ADF method (Subsection 4.2.5). The complete proposed algorithm for T-S fuzzy model identification is presented in Algorithm 4.1, and is called "Method 1".

As noted in Subsection 4.2.4, the FCM algorithm with pre-defined initial values such as the number of clusters, initial cluster centers and fuzziness weighting parameter η converges to a solution at which the objective function J in (4.10) is minimized. In practice, in many cases, randomly choosing the initial FCM parameters may cause the FCM to just obtain results which are only locally optimal [Yu *et al.*, 2004; Wu, 2012]. Furthermore, while the FCM is a semi-unsupervised method that requires the knowledge of the number of clusters, in an intelligent expert system it is desirable to use an unsupervised clustering technique. To overcome these problems, this section proposes the NUFCA [Algorithm 4.1, Steps 1-4] which uses a hybrid clustering algorithm based on two layers. NUFCA iteratively tests several values for the number of clusters c, in order to find an optimal value which is denoted as c^*. The first layer uses an unsupervised clustering technique based on KNN which tries to partition the samples into a specified number of clusters that is being varied. The second layer of NUFCA has the role of performing one iteration of the

Algorithm 4.1 Proposed T-S fuzzy model identification algorithm, which defines the NUFCA in Steps 1-4 and uses the RLS-ADF in Step 5 - Method 1.

1. Consider the output data $\mathbf{y} = [y(1), \ldots, y(L)]^T$ and construct the matrix of input data $\mathbf{X} = [x_{lj}]_{L \times N}$ in (4.9), for $l = 1, \ldots, L$, and $j = 1, \ldots, N$, using L observations;
2. Choose the degree of fuzziness $\eta > 1$; And let $\mathbf{g}_0$ be the center of data $\mathbf{X}$, and $v_t^* \leftarrow 0$;
3. Repeat the procedure below for $c = 1, 2, \ldots c_{max} = \sqrt{L}$:
   (a) Initialization for iteration c:
      i. Let $K = \left\lfloor \frac{L}{c} - 1 \right\rfloor$, and $I = \{1, 2, \ldots, L\}$, where $\lfloor \cdot \rfloor$ is the floor function;
   (b) **For** $i = 1, \ldots, c$ **construct** E_i **using** K **nearest neighbourhood:**
      i. In I find the index i of the unknown sample $\mathbf{x}^i$ which is farthest from $\mathbf{g}_{i-1}$;
      ii. $E_i = \{\mathbf{x}^i\} \cup \text{KNN}(K-1, \mathbf{x}^i)$, where $\text{KNN}(K-1, \mathbf{x}^i)$ is the set of $K-1$ nearest-neighbour samples of $\mathbf{x}^i$ that do not belong to any other already existing E_i.
      iii. Let $\mathbf{g}_i = \frac{\sum_{\mathbf{x}_k \in E_i} \mathbf{x}_k}{K}$, and $E_i \leftarrow E_i \cup \{\mathbf{g}_i\}$;
      iv. $I \leftarrow I \setminus \{i\} \setminus \text{IKNN}(K-1, \mathbf{x}^i)$, where $\text{IKNN}(K-1, \mathbf{x}^i)$ is the set of all indices n such that $\mathbf{x}^n \in \text{KNN}(K-1, \mathbf{x}^i)$;
   (c) While $I \neq \emptyset$, do:
      i. Select $r \in I$, let $I \leftarrow I \setminus \{r\}$, and calculate the distances from the still unclustered sample $\mathbf{x}^r$ to the center $\mathbf{g}_i$ of all E_i by $d(\mathbf{x}^r, \mathbf{g}_i), \forall i = 1 \ldots c$;
      ii. Assign $\mathbf{x}^r$ to the E_i with the nearest $\mathbf{g}_i$, so that $E_i \leftarrow E_i \cup \{\mathbf{x}^r\}$;
      iii. Perform the update of $\mathbf{g}_i = \frac{\sum_{\mathbf{x}_k \in E_i} \mathbf{x}_k}{K+1}$;
   (d) **Perform one iteration of FCM:**
      i. Calculate the fuzzy clustering matrix $\mathbf{U} = [\mu_{il}]_{c \times L}$ using (4.11)-(4.12) with $\mathbf{v}_i = \mathbf{g}_i$ in (4.12);
      ii. Calculate clustering validity index by (4.15) and assign it to v_t.
   (e) If $v_t > v_t^*$, then
      i. Let the optimal number of clusters be $c^* \leftarrow c$;
      ii. Let $E_i^* \leftarrow E_i$, for $i = 1, \ldots c^*$, be the optimal clustering sets;
      iii. Update the optimal clustering validity index: $v_t^* \leftarrow v_t$;
4. Using $\mathbf{U} = [\mu_{il}]_{c^* \times L}$ calculate $\mathbf{v}_i$ and σ_{ij} by (4.13)-(4.14).
5. Compute the consequent parameters $\boldsymbol{\theta}_i$, by initializing its components to small values, and then using the RLS-ADF method (Subsection 4.2.5), using recursion (4.16) for $l = 1, \ldots, L$.

FCM algorithm for each number of clusters until the antecedent parameters of the T-S fuzzy model are learned. All the parameters extracted from the first layer and the second layer are used in a fuzzy validity formulation to find the optimal number of clusters.

In the first layer of NUFCA, for each c, the initial centers of the clusters are obtained by using the KNN approach [Step 3(b)]. The basic idea in the KNN method is to try to find, from the L samples of a dataset, the k samples which have the highest levels of similarity to a specified feature vector. Specifically, in the first layer of NUFCA, the dataset $\mathbf{X}$ of (4.9) is partitioned into c clusters, in which samples of each cluster have similarity in the Euclidean distance sense, and will belong to one set E_i [Steps 3(a)-3(c)]. E_i is an auxiliary set of samples to gather the members of tentative cluster i. After all E_i sets are constructed for a certain c, then one iteration of the FCM is performed [Step 3(d)]. The final step of NUFCA consists on determining the best c, and the corresponding collection of the best E_i ($i = 1, \ldots, c$), which are termed as c^*, and E_i^* ($i = 1, \ldots, c^*$), respectively [Step 3(e)]. The results of this proposed hybrid clustering algorithm are used to set the antecedent parameters of the T-S model (4.1)-(4.2). In the final step, the algorithm uses the RLS-ADF procedure to obtain consequent the parts of T-S fuzzy model [Step 5].

4.3 Hierarchical T-S Fuzzy System Identification

The T-S fuzzy model identification which was proposed in Subsection 4.2.6 gives an efficient strategy for data fuzzy clustering while it just uses the system input/output data. A common limitation of the T-S fuzzy model identification method proposed in Subsection 4.2.6, as well as of other identification methods, is the concern regarding the selection of the correct set of input variables and associated time delays. The variable selection process is usually manual and not accompanied with the accurate selection of the right time delays, probably leading to low-accuracy results. A variable with the correct delay may contain more information about the output, than one which does not consider any delay or which considers an incorrect delay [Souza *et al.*, 2010]. This section proposes a new adaptive methodology for online learning of Takagi-Sugeno (T-S) fuzzy models based on a hierarchical particle swarm optimization (HPSO) algorithm. The HPSO design uses an automatic inputs selec-

tion strategy in a way that is able to extract the best combinations of inputs and associated delays.

The use of random initialization of the particles in a PSO may result in a very exhausting optimality search, requiring more iterations to attain convergence. Thus, in order to obtain an initial satisfactory starting point, reducing the computational cost, and to increase the algorithm's performance, NUFCA which uses a hybrid clustering algorithm based on two layers [Algorithm 4.1, Steps 1-4] is applied to initialize the particles of the proposed HPSO algorithm. While the HPSO was proposed to automatically extract all the parameters of the T-S fuzzy model from a set of input/output data, as another contribution of this work, a self-adaptive HPSO (S-AHPSO) algorithm is proposed for online identification of T-S fuzzy models.

4.3.1 Particle Swarm Optimization (PSO) Algorithm

The particle swarm optimization (PSO) algorithm was first proposed by Kennedy and Eberhart [1995]. The algorithm was attempting to simulate the choreographed motion of swarms of birds as part of a socio-cognitive study investigating the notion of collective intelligence in biological populations. PSO, unlike genetic algorithms (GA), is motivated by the simulation of social behaviour, and each candidate solution is associated with a velocity. Candidate solutions to a problem are represented by 'particles' which try to fly through the design space. The standard PSO algorithm consists of three steps, namely, generating particles positions and velocities, velocity update, and finally, position update. Here, a particle refers to a point in the design space that changes its position from one move (iteration) to another based on velocity updates. At each iteration, t, the velocity of every particle $\mathbf{v}_r^t$ will be iteratively calculated as follows:

$$\mathbf{v}_r^{t+1} = w\,\mathbf{v}_r^t + c_1 r_1(\mathbf{pbest}_r^t - \mathbf{x}_r^t) + c_2 r_2(\mathbf{gbest}^t - \mathbf{x}_r^t), \tag{4.19}$$

where $\mathbf{x}_r^t$ is the position of the particle r in iteration t, $\mathbf{pbest}_r^t$ is the best previous position of this particle (memorized by each individual particle), $\mathbf{gbest}^t$ is the best previous position among all the particles in iteration t (memorized in a common repository), w is the inertia weight, and c_1 and c_2 are positive acceleration coefficients and are known as the cognitive and social parameters, respectively. Finally, r_1 and r_2 are two random variables that take values in the range $[0, 1]$. After calculating

the velocity, the new position of every particle can be obtained as

$$\mathbf{x}_r^{t+1} = \mathbf{x}_r^t + \mathbf{v}_r^{t+1}. \tag{4.20}$$

To better address the notation of the particles inside each level of the HPSO algorithm proposed in this thesis, and also based on the case study in this work, equation (4.19) is rearranged as follows:

$$\mathbf{v}_r^p(t+1) = w_r^p(t)\mathbf{v}_r^p(t) + c_{1r}(p)r_1(\mathbf{pbest}_r^p(t) - \mathbf{x}_r^p(t)) + c_{2r}(p)r_2(\mathbf{gbest}^p(t) - \mathbf{x}_r^p(t)), \tag{4.21}$$

where, $w_r^p(t)$ and $\mathbf{pbest}_r^p(t)$ are the inertia weight and the best personal experience of particle r in iteration t inside Level p, respectively. $\mathbf{gbest}^p(t)$ is the best previous experience among all the particles inside Level p in iteration t. $c_{1r}(p)$ and $c_{2r}(p)$ are acceleration coefficients of the r-th particle in Level p. Then, (4.20) is rearranged as:

$$\mathbf{x}_r^p(t+1) = \mathbf{x}_r^p(t) + \mathbf{v}_r^p(t+1). \tag{4.22}$$

The learning process of the HPSO algorithm utilizes the conventional linearly decreasing inertia weight PSO (LDW-PSO) concept [Shi and Eberhart, 1998], which means that for each individual particle r, and iteration t:

$$w_r^p(t) = w_{r,max}^p - (w_{r,max}^p - w_{r,min}^p) \cdot t/T_1, \tag{4.23}$$

where $w_r^p(t)$ is the inertia weight in iteration t in Level p. $w_{r,max}^p$, $w_{r,min}^p$ are the initial and the final (minimum) inertia weight of particle r in Level p, respectively, and T_1 is the number of PSO iterations.

4.3.2 Hierarchical Particle Swarm Optimization (HPSO) Algorithm

This section presents HPSO, an automatic evolutionary algorithm to extract from data all the FLS parameters, and the structure, of the T-S fuzzy model (4.1)-(4.8), not requiring prior explicit expert knowledge [Rastegar *et al.*, 2017b]. The data from which the FLS is extracted is composed of a set of input observations $\mathbf{X} \in \mathbb{R}^{L\times N}$ (4.9), and a corresponding set of output observations $\mathbf{y} = [y_1, \ldots, y_L]^T \in \mathbb{R}^L$, where

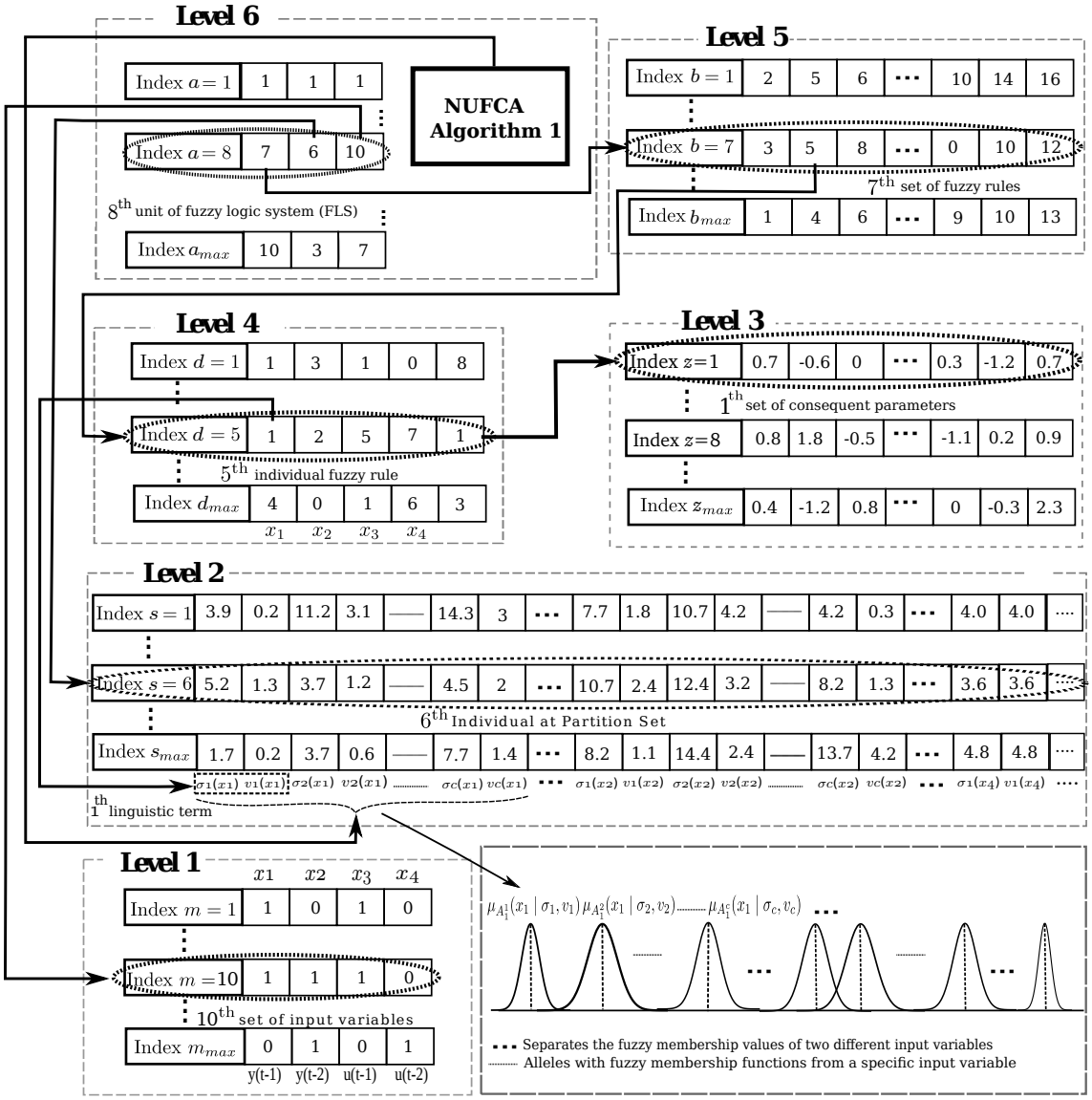


Figure 4.1: Encoding of hierarchical relations among the individuals of different levels of the HPSO algorithm that integrates the NUFCA.

y_l is the output corresponding to $\mathbf{x}_l$, for $l = 1, \ldots, L$.

The HPSO is constructed based on six hierarchical levels (Figure 4.1). The first level represents the particles of the set of input variables and their respective time delays. The particles of the second level represent all the antecedent fuzzy membership functions which constitute the fuzzy rules of the fuzzy system. Particles

at the third level represent parameters sets for the consequent parts of the fuzzy rules. The individual rule particles are defined on the fourth level, and the particles that represent sets of rules are obtained on the fifth level. Finally, the sixth level represents T-S fuzzy models, where the particles include the indices of the selected elements of the previous levels. In Algorithm 4.2, all the particles in the second, fourth, and fifth levels, and one particle of each of the first and sixth levels of the HPSO algorithm use NUFCA results to be initialized. Otherwise, all other particles are initialized randomly. The detailed description of each level is given below.

Level 1: it is composed of particles that represent the set of input variables, and respective delays. The particles of this level are represented by binary encoding, where each allele (element of the particle located in a specific position) corresponds to each input variable/delay pair (see Figure 4.1). The length of the particle is given by the total number of pairs of system variables and respective delays that are considered as possible candidates to be used as inputs for the T-S fuzzy system.

Level 2: contains Gaussian membership functions defined in the universes of discourse of the variables involved. The particle is erected by the aggregations of all Gaussian partition sets associated with the input variables. A partition set of a variable is a collection of fuzzy sets associated to the variable.

Level 3: it is constructed based on particles, where each particle represents the consequent parameters that can be used for a fuzzy rule. The length of each particle in this level is determined by the maximum number of input variables. The particle is represented by real number encoding. On each particle, the specific alleles that are taken into account for a rule are the ones that correspond to the collection of input variables selected by *Level* 1. Null values indicate the absence of consequent parameter value for the corresponding variable.

Level 4: it is formed by particles of individual rules. The length of the particle is given by the maximum number of antecedent variables plus one. The particle is represented by non-negative integer encoding, where each allele contains the index of the corresponding antecedent membership function (defined at Level 2), except the last allele in each particle which is appointed to select the indices of *Level* 3. Null values indicate the absence of membership function for the corresponding variable (i.e. the absence of the variable) and are only considered for antecedent indices.

Level 5: it is constituted by a set of fuzzy rules, where each allele contains the index of the corresponding individual rule defined in Level 4 that has been included

in the set. The particle is represented by integer encoding, where once again, null values indicate that the corresponding allele does not contribute to the inclusion of any rule in the set of fuzzy rules. The length of each particle is determined by the maximum number of fuzzy rules.

Level 6: it represents a control and specification unit for the T-S fuzzy model (TSFM). Key information required to develop and describe a TSFM design is contemplated on this level. The particle is represented by integer encoding and is constituted by three alleles. The first allele indicates the index, b, of the set of rules specified on *Level* 5. The second allele contains the s-th individual of *Level* 2 which represents the s-th collection of partition sets given by *Level* 2. The third allele represents the index, m, of the set of input variables selected on *Level* 1 (see Figure 4.1).

Level 6 is a flexible unit. Other alleles can be added to this unit in order to design a new HPSO usable with different characteristics. For example, a new extended HPSO with more complexity can be constructed for a direct fuzzy controller (DFC) design, where other alleles can be added to correspond to other important fuzzy elements such as t-norm operators, inference engine, and defuzzifier methods, and where, consequently, such additional elements of the fuzzy system can have a chance to be identified in a competition by the PSO methodology. However, as can be concluded from Subsection 4.2.1, the design of the T-S fuzzy model (as the case study of this work) is composed of: singleton fuzzifier (direct representation of a number by a singleton fuzzy set), center average defuzzifier, and the product inference engine [Wang, 1996].

The main steps of the HPSO algorithm used to learn/improve the FLS parameters are presented in Algorithm 4.2. Each level utilizes its own PSO parameters and fitness function to estimate the new position of the particles in that level. The fitness evaluation of each individual particle in each level of the hierarchy is defined as follows:

- Control unit of the fuzzy logic system (*Level* 6):
  The value of the fitness function of the a-th particle in Level 6 is defined by

$$J_6^a = \frac{1}{\text{MSE}(\mathbf{y}, \hat{\mathbf{y}}^a)}, \tag{4.24}$$

  where $\text{MSE}(\mathbf{y}, \hat{\mathbf{y}}^a) = \frac{1}{L}\sum_{l=1}^{L}(y_l - \hat{y}_l)^2$, is the mean square error of the a-th

Algorithm 4.2 Hierarchical Particle Swarm Optimization (HPSO) Algorithm.

Inputs: Set of input observations $\mathbf{X}$ (4.9), and a corresponding set of output observations $\mathbf{y} = [y_1, \ldots, y_L]^T \in \mathbb{R}^L$, where y_l is the output corresponding to $\mathbf{x}_l$, for $l = 1, \ldots, L$; Maximum number of particles for each level, a_{max}, b_{max}, d_{max}, z_{max}, s_{max}, m_{max}; Velocity limits for each level, $\mathbf{v}^p_{min} = \left[v^p_{min,i1}\right]$, $\mathbf{v}^p_{max} = \left[v^p_{max,i1}\right]$, $p = 1, \ldots, 6$; Maximum number of iterations, T_1; $w_{r,min}$, $w_{r,max}$, c_{1r}, and c_{2r} PSO parameters of the r-th particle for each level; Maximum number of particles in each level, r_{max};

Output: Optimized T-S fuzzy model (TSFM);

procedure

for all levels $p = 1, 2, \ldots, 6$ **do** // For all levels of the HPSO

for all particles $r = 1, 2, \ldots, r^p_{max}$, of Level p **do** // For all particles of the level

if $p \in \{2, 4, 5\}$ or $(p = r = 1)$ or $(p = 6$ and $r = 1)$

then Initialize $\mathbf{x}^p_r$ of p-th Level using the results extracted with NUFCA [Algorithm 4.1, Steps 1-4];

else Randomly initialize $\mathbf{x}^p_r$ **endif**

Initialize $\mathbf{v}^p_r$ of p-th Level randomly within the velocity range of $(\mathbf{v}^p_{min}, \mathbf{v}^p_{max})$;

$\mathbf{pbest}^p_r \leftarrow \mathbf{x}^p_r$;

end for

end for

Evaluate all particles using one of the equations (4.24)-(4.27), according to the level of each particle;

Identify the $\mathbf{gbest}^p$ of the swarm p, for $p = 1, \ldots, 6$;

$t \leftarrow 1$;

while $t \leq T_1$ **do**

for all levels $p = 1, 2, \ldots, 6$ **do**

for all particles $r = 1, 2, \ldots, r^p_{max}$, of Level p **do**

Compute $w^p_r(t)$ and $\mathbf{v}^p_r(t)$ using (4.23) and (4.21), respectively;

Update $\mathbf{x}^p_r(t)$ using (4.22);

end for

end for

Evaluate $J^{(r)}_p$, for each particle $r = 1, 2, \ldots, r^p_{max}$, of each level $p = 1, 2, \ldots, 6$, using (4.24)-(4.27);

Update $\mathbf{pbest}^p_r(t)$ and $\mathbf{gbest}^p(t)$ based on the $J^{(r)}_p$ results from (4.24)-(4.27), for $p = 1, 2, \ldots, 6$, $r = 1, 2, \ldots, r^p_{max}$;

$t \leftarrow t + 1$;

end while

end procedure

fuzzy system ($a = 1, \ldots, a_{max}$), y_l is the l-th output observation (target), and $\hat{y}_l$ is the corresponding output value predicted by the FLS;

- Units of *Levels* 1, 2, *and* 5:
  The value of the fitness function of the u-th particle in Level e, for $e = 1, 2, 5$, is evaluated by
  $$J_e^u = \max\left(J_6^{f_{e,1}}, \cdots, J_6^{f_{e,p(e)}}\right), \tag{4.25}$$
  where $\{f_{e,1}, \ldots, f_{e,p(e)}\} \subseteq \{1, \ldots, a_{max}\}$ is the subset of all the particles of Level 6 that contain the u-th individual of Level e (see Figure 4.1);

- Individual rule (*Level* 4):
  The value of the fitness function of the d-th particle in Level 4 is defined by
  $$J_4^d = \max(J_6^{r_1}, \ldots, J_6^{r_q}), \tag{4.26}$$
  where $\{r_1, \ldots, r_q\} \subseteq \{1, \ldots, a_{max}\}$ is the subset of all the particles of Level 6 that involve the d-th individual rule of Level 4 (indirect involvement through Level 5, see Figure 4.1);

- Consequent sets (*Level* 3):
  The value of the fitness function of the z-th particle in Level 3 is defined by
  $$J_3^z = \max(J_6^{g_1}, \ldots, J_6^{g_p}), \tag{4.27}$$
  where $\{g_1, \ldots, g_p\} \subseteq \{1, \ldots, a_{max}\}$ is the subset of all the particles of Level 6 that involve the z-th consequent set of Level 3 (indirect involvement through Levels 5 and 4, see Figure 4.1).

Each level of the particles' hierarchy is evolved separately as an independent PSO algorithm using its own particles, its own fitness function, and its own PSO parameters. However, since the values of the fitness evaluations of the particles in each level also depend on the particles in other levels, then the evolution of each level is also influenced by the evolution of all the other levels.

4.3.3 Adaptive Online the T-S Fuzzy Modelling

The results extracted by the HPSO methodology, Subsection 4.3.2, and Algorithm 4.2 will be used to initialize the T-S fuzzy model for the operation in online mode. For the online process the consequent parameters vector $\mathbf{\Theta}$ (4.7) is denoted by $\mathbf{\Theta}_{on}$, and is self-adjusted, being updated as follows whenever a new data sample becomes available at time instant k:

$$\mathbf{\Theta}_{on}(k) = \mathbf{\Theta}_{on}(k-1) + \Delta\mathbf{\Theta}_{on}(k), \tag{4.28}$$

where $\mathbf{\Theta}_{on}$ is initialized by the consequent parameters that have resulted from Algorithm 4.2, and $\Delta\mathbf{\Theta}_{on}$ is calculated for each incoming sample at each time instant k using the S-AHPSO algorithm as described in this sub-section.

Although PSO has shown some important advances by providing high speed of convergence in specific problems, it does exhibit some shortages. It was found that PSO has a poor ability to search at a fine grain scale because it lacks an adequate velocity control mechanism. Many approaches have been attempted to improve the performance of conventional PSO by variable inertia weight [Ratnaweera *et al.*, 2004]. The inertia weight, which balances the global exploration and local exploitation abilities of the swarm, is critical for the performance of PSO. A big inertia weight facilitates exploration, but it makes the particle to take a long time to converge. Conversely, a small inertia weight makes the particle to converge fast, but it some times leads to local optima. The performance of PSO can be improved if an appropriate adaptive strategy for the inertia weight [Ratnaweera *et al.*, 2004] replaces typical inertia weight strategies like the conventional linearly decreasing inertia weight PSO (LDW-PSO) [Shi and Eberhart, 1998]. For example, a variable inertia weight can be formulated to depend on the particle's position or velocity.

Motivated by these facts, this section proposes an adaptive process for obtaining the inertia weight. For each newly available data sample at time instant k, S-AHPSO runs PSO iterations $t = 1, \ldots, T_2$. For each iteration t of S-AHPSO, the inertia weight w_r for each particle r is obtained from feedback taken from the best

memories of the individual particle and of the swarm as follows:

$$w_r(t) = w_{r,min} + (w_{r,max} - w_{r,min}) \times \exp\left(-\frac{\gamma}{|\, y^{(\mathbf{pbest_r})}(t) - y^{(\mathbf{gbest})}(t)\,| + \beta} \times \frac{t-1}{T_2 - t + \delta(t - T_2)} \right) [1 - \delta(t - T_2)], \tag{4.29}$$

where γ and β are positive constants, $\delta(k)$ is the Kronecker Delta function, where $\delta(k) = 0$, for all $k \in \mathbb{N}$, except $\delta(0) = 1$, and T_2 is the number of S-AHPSO iterations in t at each time instant k. Also, $y^{(\mathbf{pbest}_r)}(t)$ is the result obtained by the best previous position of the r-th particle for all non-future values of t for the current k, and $y^{(\mathbf{gbest})}(t)$ is the best output estimation attained among all particles of the swarm, until the current iteration t for the current k.

If the effect of $\left| y^{(\mathbf{pbest_r})}(t) - y^{(\mathbf{gbest})}(t) \right|$ is not considered in (4.29), then, as can be seen, the inertia weight in both (4.23) and (4.29) follows a decreasing tendency. On the other hand, if a particle r is far away from the swarm, then the term $\left| y^{(\mathbf{pbest_r})}(t) - y^{(\mathbf{gbest})}(t) \right|$ in the denominator in (4.29) will reduce the decreasing tendency of the particles' inertia weights, and thus increase the degree of exploration in order to give better possibilities for particle r to join the swarm.

For updating of consequent rules in order to perform the online construction of the T-S fuzzy model, Eq. (4.19) is reformulated as:

$$\mathbf{v}_r(t+1) = w_r(t)\mathbf{v}_r(t) + c_{1r}r_1(\mathbf{pbest}_r(t) - \Delta\boldsymbol{\Theta}_r(t)) + c_{2r}r_2(\mathbf{gbest}(t) - \Delta\boldsymbol{\Theta}_r(t)), \tag{4.30}$$

and the new position of every particle can be obtained as

$$\Delta\boldsymbol{\Theta}_r(t+1) = \Delta\boldsymbol{\Theta}_r(t) + \mathbf{v}_r(t+1). \tag{4.31}$$

The complete proposed S-AHPSO online adaptive system identification algorithm is presented in Algorithm 4.3. For each new data sample that becomes available at time instant k, when the S-AHPSO algorithm terminates the PSO iterations $t = 1, \ldots, T_2$ for that sample k, the final $\Delta\boldsymbol{\Theta}_r(t)$ value is used as $\Delta\boldsymbol{\Theta}_{on}(k)$ in (4.28). The S-AHPSO algorithm, requires the use of the HPSO algorithm to generate its inputs, and is also called "Method 2".

Algorithm 4.3 Self-adaptive Hierarchical Particle Swarm Optimization (S-AHPSO) Algorithm, which requires the use of the HPSO algorithm to generate the inputs - Method 2.

Input: The T-S fuzzy system learned by the HPSO Algorithm 4.2 (input variables, antecedent parameters, consequent parameters, the fuzzy rules, and the final learned model parameters).
Output: $\Delta\boldsymbol{\Theta}_{on}(k)$, for each new data sample that becomes available during the online operation at time instant k;

1. Initialization:
   (a) Select r_{max}, the maximum number of particles for the swarm $\Delta\boldsymbol{\Theta}_r$; as well as the maximum number of iterations T_2; Select the velocity limits $\mathbf{v}_{min} = [v_{min,i1}]$, $\mathbf{v}_{max} = [v_{max,i1}]$;
   (b) Design the PSO parameters $w_{r,max}$, $w_{r,min}$, c_{1r}, c_{2r} for each particle r of the swarm $\Delta\boldsymbol{\Theta}_r$;
   (c) Initialize $[\boldsymbol{\Theta}_{on}]_{(c\,N)\times 1}$ of the T-S fuzzy model using the best result obtained from Level 3 of Algorithm 4.2, where $[\boldsymbol{\Theta}_{on}]_{(c\,N)\times 1}$ is the consequent parameters vector and c is the optimal number of individual fuzzy rules which have both resulted from Algorithm 4.2, and N is the maximum number of selected input variables, respectively;
2. For/using each newly arriving online sample k, do:
   for all particles of the swarm, $r = 1, 2, \ldots, r_{max}$ **do**
   Randomly initialize $[\Delta\boldsymbol{\Theta}_r]_{(c\,N)\times 1}$;
   Initialize $\mathbf{v}_r$ randomly within the velocity range $(\mathbf{v}_{min}, \mathbf{v}_{max})$;
   Let $\mathbf{pbest}_r \leftarrow \Delta\boldsymbol{\Theta}_r$;
   end for
   Evaluate each particle of the swarm $\boldsymbol{\Delta\Theta}_r$ using the fitness function $J^{(r)}$ (4.32);
   Identify the best particle **gbest**;
   for all $t = 1, \ldots, T_2$ **do**
   Compute $w_r(t)$ and $\mathbf{v}_r(t+1)$ using (4.29) and (4.30), respectively;
   Update $\Delta\boldsymbol{\Theta}_r(t)$ using (4.31);
   Calculate the fitness function of each particle of the swarm $\Delta\boldsymbol{\Theta}_r$ using the fitness function $J^{(r)}$ (4.32);
   Update $\mathbf{pbest}_r$ and **gbest**;
   end for
   Let $\Delta\boldsymbol{\Theta}_{on}(k) \leftarrow \Delta\boldsymbol{\Theta}_r(t)$;
   Adapt the T-S fuzzy model parameters ($\boldsymbol{\Theta}$ of Eq. (4.7)) using (4.28) ;

The numbers of iterations T_1 and T_2 of Algorithms 4.2, and 4.3, respectively, could be seen as the maximum numbers of iterations if other/early termination conditions (e.g. as a function of fitness) would be introduced in these algorithms.

The fitness function applied in Algorithm 4.3 is

$$J^{(r)} = 1/\text{MSE}(r) = \left[(y_{re}(k) - \hat{y}^{(r)}(k))^2\right]^{-1}, \tag{4.32}$$

where $\text{MSE}(r)$ is the mean square error of the r-th particle, $y_{re}(k)$ is the target output value at instant k, and $\hat{y}^{(r)}(k)$ is a prediction of $y_{re}(k)$ calculated by the r-th particle using (4.28) and (4.2).

4.4 Experimental Results

This section presents simulation and real-world experimental results to demonstrate the feasibility, performance, and effectiveness of the proposed T-S design methodologies in system identification. The performances of the proposed adaptive fuzzy identification methodologies are demonstrated on three setups: a real WWTP plant, a simulated WWTP plant (Benchmark simulation model 1 (BSM1)), and a real-world experimental setup composed of two coupled DC motors. For the real-world two coupled DC motors, and the simulated WWTP plant the proposed identification methodologies are analyzed and quantitatively compared with the recursive partial least squares (RPLS) [Dayal and MacGregor, 1997] adaptive approach, and with a new fuzzy c-regression model algorithm (NFCRMA) [Li *et al.*, 2009]. Furthermore, the results obtained with five more methods are compared: a fuzzy modeling based on MLP Neural Networks (MLPNN) [Nelles, 2001], a recent proposed incremental local learning soft sensing algorithm (ILLSA) for adaptive soft sensors [Kadlec and Gabrys, 2011], a system Identification using adaptive PSO (APSO) [Alfi and Modares, 2011], an extreme learning machine for regression (ELM) [Huang *et al.*, 2012], and a hierarchical genetic approach (HGA) for learning T-S fuzzy models [Mendes *et al.*, 2012]. Method 1 and Method 2 were defined in Sections 4.2.6, and 4.3.3, respectively. Following such definitions, in all simulations presented below in this chapter, "Method 1" indicates the T-S fuzzy model identification methodology specified in Algorithm 4.1, while the learned consequent parameters of the T-S fuzzy model are updated by the RLS-ADF methodology (Subsection 4.2.5); And "Method

Table 4.1: Variables of the wastewater treatment plant dataset.

Variables	Description
u_1	Amount of chlorine in the influent;
u_2	Amount of chlorine in the effluent;
u_3	Turbidity in the raw water;
u_4	Turbidity in the influent;
u_5	Turbidity in the effluent;
u_6	Ph in the raw water;
u_7	Ph in the influent;
u_8	Ph in the effluent;
u_9	Color in the raw water;
u_{10}	Color in the influent;
u_{11}	Color in the effluent;
y	Flour in the effluent.

2" refers to the adaptive online T-S fuzzy identification methodology given by the HPSO Algorithm 4.2, while the learned consequent parameters of the T-S fuzzy model are updated by the S-AHPSO specified in Algorithm 4.3.

4.4.1 Application to a Real Wastewater Treatment System

In this section, the performances of the two proposed identification methodologies (Methods 1 and 2) are studied. Specifically, a Soft Sensor application is studied. The objective of this experiment is to estimate the flour concentration in the effluent of a real-world urban wastewater treatment plant (WWTP). The dataset of plant variables that is available for learning consists of 11 input variables, $u_1, \ldots, u_{11}$, and one target output variable to be estimated, y. The variables correspond to physical values, such as pH, turbidity, color of the water and others. The input variables are measured on-line by plant sensors, and the output variable in the dataset is measured by laboratory analysis. The plant variables are described in Table 4.1. The historical data set comprises three years of acquisition, with a sample rate of 2 [hours] for the variables acquired by sensors (input variables). The target variable, the fluorine, is laboratory measured at every 24 [hours]. Complete information about the WWTP can be found in [Souza *et al.*, 2013; Souza and Araújo, 2014].

To construct the dataset, the first three delayed versions of each variable were chosen as candidates for inputs of the T-S fuzzy model. Specifically, the following

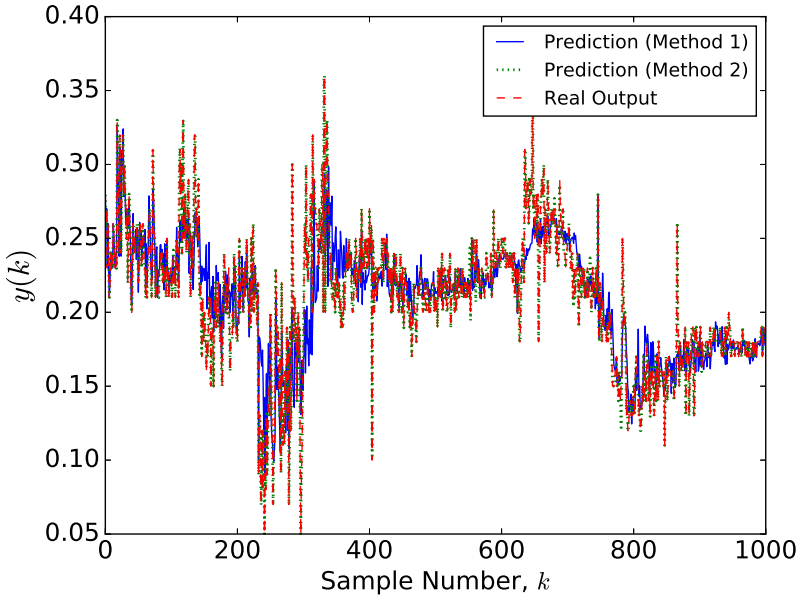


Figure 4.2: Modeling performance of the proposed system identification of Method 1 (learned by Algorithm 4.1 and updated by the RLS-ADF method), and Method 2 (learned by Algorithm 4.2 and updated by Algorithm 4.3) for the real wastewater treatment system data set.

combinations of process variables and delays are used as the candidates for inputs of the T-S fuzzy model to predict $y(t)$: $[u_1(t-1), u_1(t-2), u_1(t-3), \ldots, u_{11}(t-1), u_{11}(t-2), u_{11}(t-3)]$. The available data set was split into 30% of data for training, and the remaining 70% of data was used to test the proposed algorithm. The selected degree of fuzziness was set to $\eta = 2$, and the optimal number of clusters that resulted from Algorithm 4.1 was $c^* = 13$. Figure 4.2 shows the predicted values that resulted from the application of Method 1 and Method 2, as well as the desired (real) values of the target variable to be estimated, for the WWTP experiment. As can be seen, the accuracy of the modeling is acceptable for both proposed T-S fuzzy modeling methodologies. Numerical results comparing the performance of the proposed identification methodologies and the works RPLS [Dayal and MacGregor, 1997], MLPNN [Nelles, 2001], NFCRMA [Li *et al.*, 2009], APSO [Alfi and Modares, 2011], ILLSA [Kadlec and Gabrys, 2011], ELM [Huang *et al.*, 2012], and HGA [Mendes *et al.*, 2012] are presented in Table 4.2.

As can be seen comparing with other methods, larger values of the fitness func-

Table 4.2: Comparison results on the test dataset for the real wastewater treatment plant (WWTP).

Method	Number of rules	Number of inputs	Inputs	1/MSE	CT [minutes]
RPLS [Dayal and MacGregor, 1997]	-	-	All variables	840.9	2.18
MLPNN [Nelles, 2001]	-	-	All variables	424.3	8.33
NFCRMA [Li *et al.*, 2009]	-	-	All variables	206.1	5.24
APSO [Alfi and Modares, 2011]	-	-	All variables	978.5	7.15
ILLSA [Kadlec and Gabrys, 2011]	-	-	All variables	1197.6	3.68
ELM [Huang *et al.*, 2012]	-	-	All variables	419.5	2.92
HGA [Mendes *et al.*, 2012]	20	15	$u_1(t-2)$, $u_2(t-3)$, $u_3(t-2)$, $u_3(t-3)$, $u_4(t-1)$, $u_4(t-2)$, $u_4(t-3)$, $u_5(t-2)$, $u_5(t-3)$, $u_6(t-3)$, $u_7(t-1)$, $u_7(t-2)$, $u_8(t-3)$, $u_{10}(t-2)$, $u_{11}(t-1)$	441.6	13.07
Method 1	13	11	$u_1(t-1)$, $u_1(t-3)$, $u_3(t-1)$, $u_6(t-3)$, $u_7(t-1)$, $u_7(t-3)$, $u_8(t-1)$, $u_8(t-3)$, $u_9(t-3)$, $u_{10}(t-1)$, $u_{11}(t-3)$	1716.2	2.45
Method 2	28	15	$u_2(t-1)$, $u_3(t-1)$, $u_4(t-2)$, $u_4(t-3)$, $u_5(t-2)$, $u_6(t-1)$, $u_7(t-2)$, $u_7(t-3)$, $u_8(t-1)$, $u_8(t-2)$, $u_9(t-1)$, $u_9(t-2)$, $u_{10}(t-1)$, $u_{10}(t-3)$, $u_{11}(t-1)$	2120	7.88

tion ($1/MSE$), $MSE = (1/L)\sum_{l=1}^{L}(y_k - \hat{y}_k)^2$, in the test dataset are obtained with the two methods proposed in this work, where y_k and $\hat{y}_k$ are the real and predicted values of y at instant k, respectively. The computational time (CT) of each of the used methods is given in Table 4.2. The computational time for Method 2 was 7.88 [minutes], while for Method 1 it was 2.45 [minutes]. Comparing with Method 1, a larger value of the fitness function was obtained by Method 2. However, comparing with Method 1, more computational time was spent by the Method 2 to be run. The implementation of Method 2 uses the NUFCA results which is a part of Method 1. Figure 4.3 illustrates trajectory of fitness evaluation in the WWTP for both Method 1 and Method 2, when the number of clusters considered in the input WWTP data was changed. The first purpose for designing the NUFCA was to address concerns with the T-S fuzzy modelling based on FCM. However, the result in Figure 4.3 show that finding the optimal number of clusters can improve the estimation of the WWTP plant. This parameter plays an important role in the

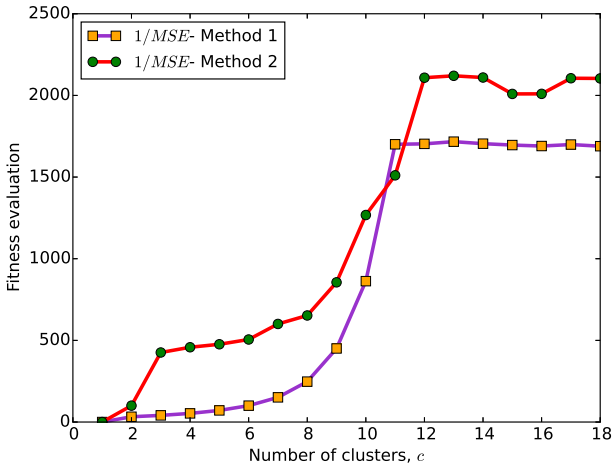


Figure 4.3: Fitness evaluation that resulted from the application of Method 1 and Method 2 on the real WWTP prediction experiment, against the number of fuzzy clusters on the NUFCA.

design of the fuzzy membership functions, and individual rules. As can be seen in the fitness evaluation trajectory obtained by Method 1, in the range from $c = 2$ to $c = 11$ the fitness evaluation values are improving. After $c = 11$ the optimal trajectory evolves with small changes. However, for this case study, the best result was obtained for $c^* = 13$. The fitness evaluation that resulted from Method 2 shows a different trajectory when comparing to the trajectory of fitness values that resulted resulted from Method 1. For example after $c = 11$ the optimal trajectory on Method 2 evolves with more changes than on Method 1. However, like Method 1, the best fitness value obtained by Method 2 was obtained for $c^* = 13$.

4.4.2 General Characteristics of the Benchmark Simulation Model 1 (BSM1)

WWTPs are industrial structures designed to remove biological or chemical waste products from water. They are complex nonlinear systems subject to large disturbances in influent flow rate and pollutant load, together with uncertainties concerning the composition of the incoming wastewater [Belchior *et al.*, 2012]. The Benchmark Simulation Model no.1 (BSM1) is a platform-independent simulation

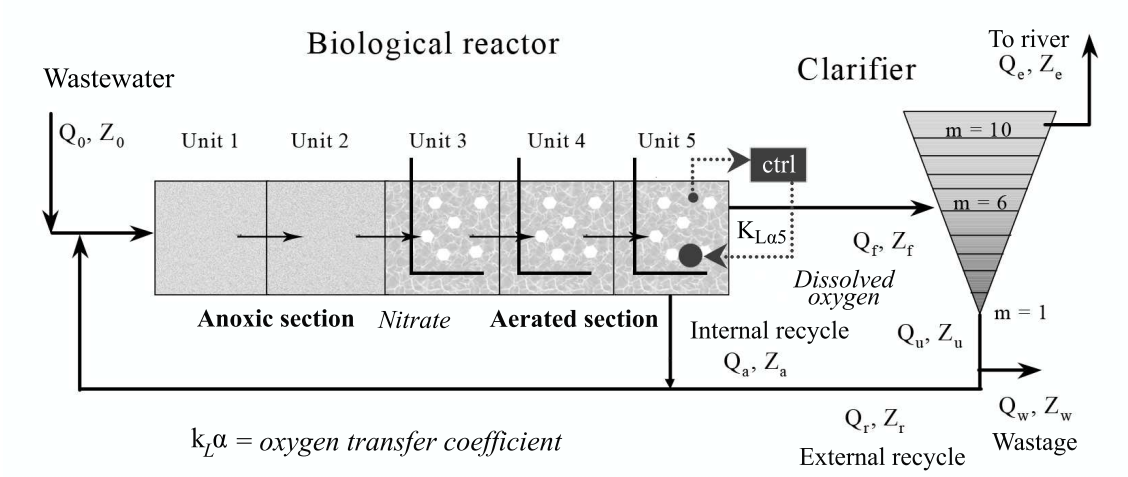


Figure 4.4: General overview of the BSM1 plant [Belchior *et al.*, 2012].

environment of WWTPs which has been undertaken in Europe by Working groups of COST Actions 682 and 624 that are dedicated to the optimization of performance and cost-effectiveness of wastewater management systems [Jeppsson and Pons, 2004]. This development work is now continued under the umbrella of the IWA Task group on Benchmarking of Control Strategies for WWTPs. The T-S fuzzy model learning architectures proposed in this chapter are tested in the BSM1. A general overview of the BMS1 plant is presented in Figure 4.4. The benchmark plant is composed of a five compartment activated sludge reactor consisting of two anoxic tanks followed by three aerobic tanks. Each compartment is characterized by flow rate Q_k, concentration Z_k, volume V_k, and reaction rate r_k. Volumes V_k for non-aerated compartments are $V_1 = V_2 = 1.000\,[\mathrm{m}^3]$, and for aerated compartments are $V_3 = V_4 = V_5 = 1.333\,[\mathrm{m}^3]$. Compartments 3-4 have a fixed oxygen transfer coefficient ($K_{La} = 10\,[\mathrm{h}^{-1}] = 240\,[\mathrm{days}^{-1}]$) while in compartment 5, the dissolved oxygen (DO) concentration, D_{OC}, is controlled by manipulation of the K_{La5} oxygen transfer coefficient. For more details about the BSM1 plant, references [Jeppsson and Pons, 2004; Belchior *et al.*, 2012; Mendes *et al.*, 2014] are recommended. The sampling period is 15 [min], and the simulations have a maximum of 14 [days].

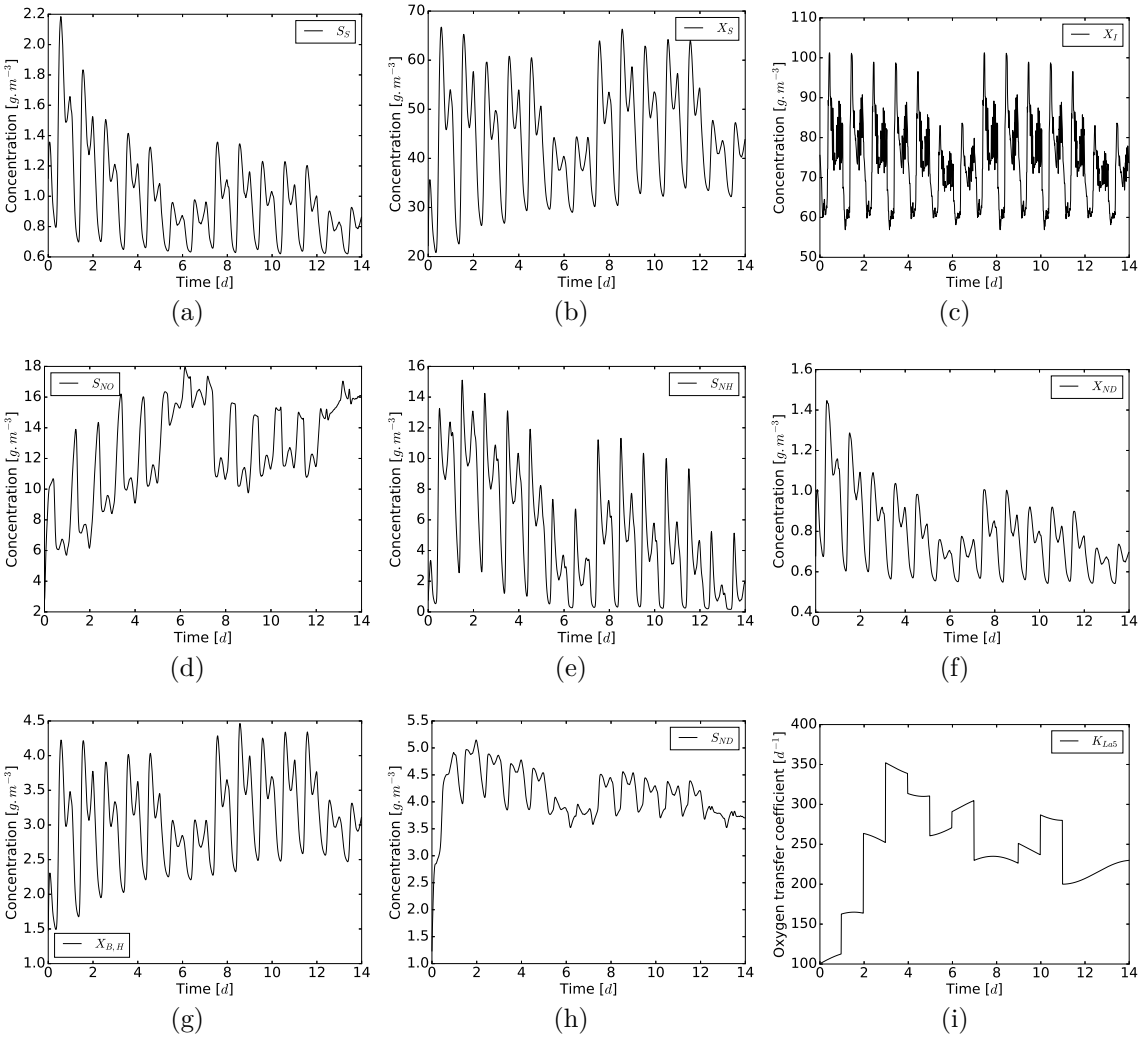


Figure 4.5: Input variables of the data set applied for learning of the HPSO and algorithm 4.1.

4.4.3 Application to the Benchmark Simulation Model 1 (BSM1)

For the learning process of the proposed algorithms, a data set has been gathered from the plant operation and consists of 10 main variables $(u_1, \ldots, u_{10})$, where 9 of them are input variables $(u_1, \ldots, u_9) = (S_S, X_S, X_I, S_{NO}, S_{NH}, X_{ND}, S_{ND}, X_{B,H}, K_{La5})$, and the remaining variable is the target output variable to be estimated, $y = u_{10} = D_{OC}$. Figure 4.5 illustrates the input variables of the data set. The plant variables are described in Table 4.3.

Table 4.3: List of BSM1 variables.

Notation	Description
S_S	Readily biodegradable substrate;
X_S	Slowly biodegradable substrate;
X_I	Particulate inert organic matter;
$X_{B,H}$	Active heterotrophic biomass;
$X_{B,A}$	Active autotrophic biomass;
X_P	Particulate products arising from biomass decay;
S_O	Oxygen;
S_I	Soluble inert organic matter;
S_{NO}	Nitrate and nitrite nitrogen;
S_{NH}	$NH4^+ + NH3$ nitrogen;
S_{ND}	Soluble biodegradable organic nitrogen;
X_{ND}	Particulate biodegradable organic nitrogen;
K_{La}	Oxygen transfer coefficient;
S_{ALK}	Alkalinity;
D_{OC}	Dissolved oxygen concentration.

To construct the data set, the first three delayed versions of each of the variables $u_1, \ldots, u_9$, were chosen as candidates for inputs of the T-S fuzzy model. Specifically, the following combinations of process variables and delays were used as the candidates for inputs of the T-S fuzzy models to predict $y(t)$: $[u_1(t-1), u_1(t-2), u_1(t-3), \ldots, u_9(t-1), u_9(t-2), u_9(t-3)]$. The total data set includes 1344 samples where the first 480 samples are used as the training data set to learn the T-S fuzzy model parameters by Algorithm 4.1 in Method 1, and by the HPSO algorithm, i.e. Algorithm 4.2, in Method 2. The remaining 864 samples are used as the test data set for RLS-ADF (see Subsection 4.2.5) in Method 1, and for Algorithm 4.3 in Method 2.

The selected degree of fuzziness was set to $\eta = 2$ in Algorithm 4.1. The PSO parameters $w_{r,max} = 1$, $w_{r,min} = 0.3$, $c_{1r} = 1.5$, and $c_{2r} = 2$ are used in common among all particles in both Algorithm 4.2 and Algorithm 4.3. In Algorithms 4.2 and 4.3, all components of the minimum and maximum velocities are $v^p_{min,i1} = v_{min,i1} = -2$, and $v^p_{max,i1} = v_{max,i1} = 2$, respectively. The maximum number of iterations of the HPSO algorithm is $T_1 = 200$. For the S-AHPSO algorithm, $\gamma = 1$, $\beta = 0.1$, and a maximum number of iterations $T_2 = 50$ were considered. The numbers of particles for each level of the HPSO architecture are: $a_{max} = b_{max} = 30$, $d_{max} = z_{max} = 50$, $s_{max} = 40$, and $m_{max} = 20$. Also, $r_{max} = 25$ is considered for

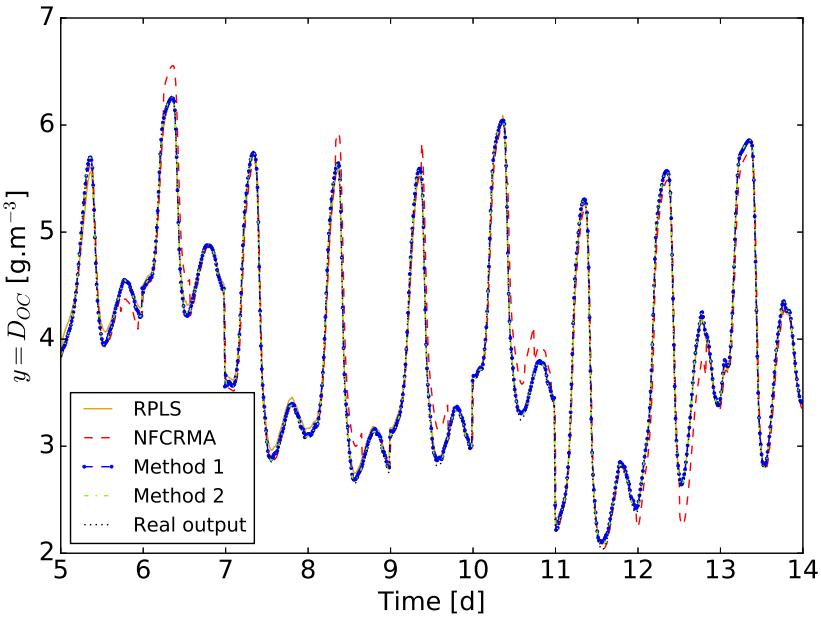


Figure 4.6: Modeling performance of the proposed system identification methodologies of Method 1, and Method 2 for the $y = D_{OC}$ variable of the BSM1 wastewater treatment system data set.

the S-AHPSO architecture. The parameters aforementioned in this paragraph were tuned by the designer. On the other hand, the optimal number of clusters c^* is self-adjusted by Algorithm 4.1, which obtained a value of $c^* = 10$. Also, the consequent parameters are self-adjusted, being calculated by the RLS-ADF for Method 1, and by Algorithm 4.3 for Method 2. The following parameters were set for the RLS-ADF method: $\rho = 0.999$, $\varphi_i = 1$, $\tau_i = \nu_i = 1 \times 10^{-8}$, for $i = 1, \ldots, c^*$.

Figure 4.6 illustrates the predicted values obtained by the proposed methodologies and the desired (real) values of the target variable $y = D_{OC}$ to be estimated, for the data set of the BSM1 WWTP experiment. Numerical results, with the best fitness function, comparing the performance of the proposed methodologies and the works FCM-FRLS [Kulhavý, 1987], RPLS [Dayal and MacGregor, 1997], MLPNN [Nelles, 2001], NFCRMA [Li *et al.*, 2009], APSO [Alfi and Modares, 2011], ELM [Huang *et al.*, 2012], and HGA [Mendes *et al.*, 2012] are presented in Table 4.4.

For a test dataset with L samples, define $J_{best} = \frac{1}{\text{MSE}}$ as the fitness function, and $\text{MSE} = \sum_{k=1}^{L}(y(k) - y_{re}(k))^2$, where $y(k)$ and $y_{re}(k)$ are the estimated and real plant output values at time instant k, respectively. The main objective is to

Table 4.4: Comparison results on the test data set for the wastewater treatment plant Benchmark Simulation Model 1 (BSM1).

Method	Number of rules	Number of inputs	Inputs	1/MSE	CT [minutes]
FCM-FRLS	-	-	All variables	1483.7	2.803
RPLS	-	-	All variables	119.3	4.375
MLPNN	-	-	All variables	1226.9	16.247
NFCRMA (Random initialization)	-	-	All variables	59.2	10.36
NUFCA-NFCRMA	-	-	All variables	373.7	10.55
APSO (FCM initialization)	-	-	All variables	135.9	13.75
NUFCA-APSO	-	-	All variables	1024.6	14.88
ELM	-	-	All variables	46.7	5.953
HGA	32	13	$u_1(t-2)$, $u_2(t-1)$, $u_2(t-2)$, $u_3(t-1)$, $u_3(t-3)$, $u_4(t-3)$, $u_5(t-1)$, $u_6(t-1)$, $u_7(t-1)$, $u_7(t-3)$, $u_8(t-1)$, $u_9(t-1)$, $u_9(t-2)$	1349.5	24.66
Method 1	10	9	$u_1(t-3)$, $u_2(t-1)$, $u_3(t-1)$, $u_3(t-3)$, $u_4(t-1)$, $u_5(t-2)$, $u_7(t-1)$, $u_9(t-2)$, $u_9(t-3)$	1721.2	4.35
Method 2	20	8	$u_2(t-1)$, $u_3(t-1)$, $u_3(t-3)$, $u_4(t-1)$, $u_5(t-2)$, $u_7(t-1)$, $u_9(t-2)$, $u_9(t-3)$	2247.2	13.07

obtain a large value of J_{best} which corresponds to the goal of having a small MSE. Based on the results in Table 4.4, the largest value of the fitness function in the test data set is obtained with Method 2. Also, comparing with the HGA, the proposed methodologies (Methods 1 and 2) use a lower number of variables and fuzzy rules, but show a better prediction performance and faster learning performance. The computational time (CT) of each of the used methods is given in Table 4.4. With an Intel(R) core (TM) i7-2600 CPU at 3.4 GHz, the simulation times for Method 1 and Method 2 were 17.6%, and 53% of the simulation time of the HGA, respectively. Also, from the results in Table 4.4 it is seen that 8 input variables $u_2(t-1)$, $u_3(t-1)$, $u_3(t-3)$, $u_4(t-1)$, $u_5(t-2)$, $u_7(t-1)$, $u_9(t-2)$, and $u_9(t-3)$, were selected by the HPSO algorithm (and therefore also used in Method 2) (among all the 27 the input (variable, delay) pairs candidates that were considered) for the T-S fuzzy model of the BSM1 process that was learned. Also, for Method 1 (Algorithm 4.1), 9 input variables $u_1(t-3)$, $u_2(t-1)$, $u_3(t-1)$, $u_3(t-3)$, $u_4(t-1)$, $u_5(t-2)$, $u_7(t-1)$, $u_9(t-2)$, and $u_9(t-3)$ were chosen for the learning process. The membership

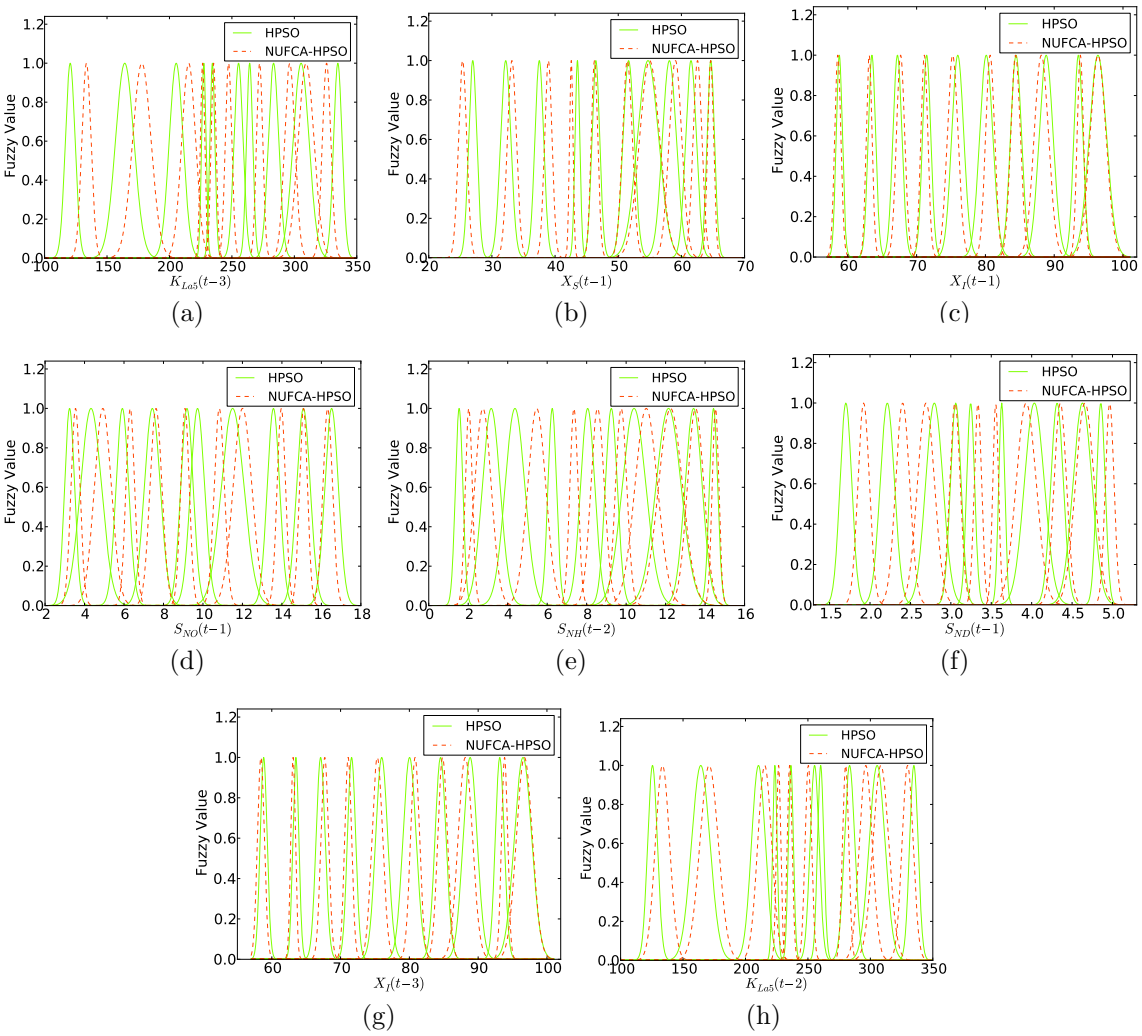


Figure 4.7: Membership functions of the partition set of the selected variables obtained by the HPSO method with random initialization and by using the NUFCA as the initialization algorithm.

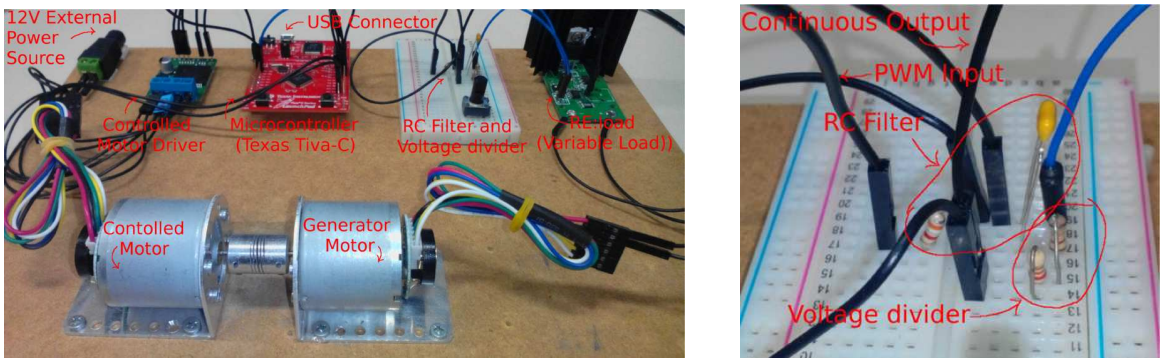
functions extracted by Level 2 for the variables selected by Level 1 for the whole T-S fuzzy model obtained by the HPSO with random initialization, and by the HPSO initialized by the NUFCA algorithm (as specified in Algorithm 4.2) are shown in Figure 4.7.

Some points related to the methods' performance results presented in Table 4.4 are worth noting. The MLPNN method was run based on two different types of iterations. The method has an external iterative cycle, with a maximum number of 1000 iterations for this example which is considered as a stop criterion for the MLPNN.

Also, the MLPNN learning is done by the Levenberg-Marquardt method which has an inner additive iteration to update neural network weights. Updated neural network weights which result from the inner iteration will initialize the weights of the next external iteration. The implementation of the HGA was also accompanied with more iterations than was considered firstly. The method was initially programmed for 1000 iterations. However, due to the random initialization, the result of the HGA method just depicted after 100 times runs of the complete algorithm with 1000 iterations each. The result in Table 4.4 is the best result among all repetitions of the HGA. The performance of the APSO method [Alfi and Modares, 2011] has shown a variety of sensitivity phenomena to some parameters inside the APSO algorithm. The best result was obtained just after all observations on the APSO's performance variations resulting from the variations of the aforementioned significant parameters. However, the best fitness evaluation obtained by Method 2 (Algorithms 4.2 and 4.3), and shown in Table 4.4, is simply obtained with just one test of the proposed method with $T_1 = 200$ and $T_2 = 50$. Furthermore, the APSO's initialization was performed with the presented NUFCA method. A final point should be mentioned regarding to the efficiency of NUFCA method: based on the results in Table 4.4, the efficient role of the NUFCA to enhance the performance of other regressors is also remarkable. The best fitness evaluation was obtained by Method 2, while in the implementation of Method 2 the results of the NUFCA operation/performance have been used. As it was seen before, NUFCA is also a part of Method 1.

4.4.4 Real-World Control of Two Coupled DC Motors

The real plant from Figure 4.8(a) is composed of two motors which are coupled by their shafts [Mendes *et al.*, 2016; Rastegar *et al.*, 2017d,c]. The first motor is controlled by Pulse Width Modulation (PWM) and makes the second motor rotate and behave as a generator. On the other hand, this generator is connected to a variable/controllable electrical load, $R_L(k)$, which acts as an energy sink where the amount of dissipated energy can be varied. The selected motors have an encoder with 64 Counts Per Revolution (CPR) and are rated for 11000 Rotations Per Minute at 12V (free run). The velocity units are [pp/(100 ms)] (pulses per 100 [ms]). The controlled motor is powered by a motor driver that drains energy from an external power source and receives PWM control signals. The motor that works as a gener-



(a) (b)

Figure 4.8: (a) The experimental scheme of the two coupled DC motors, and (b) RC filtering and the voltage divider.

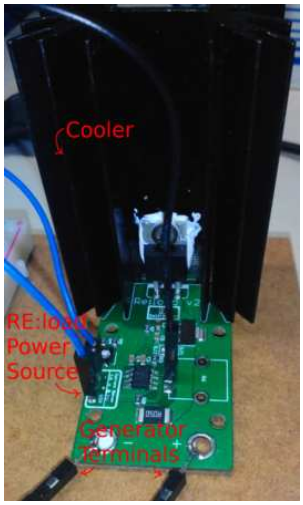


Figure 4.9: Variable load (Re:load).

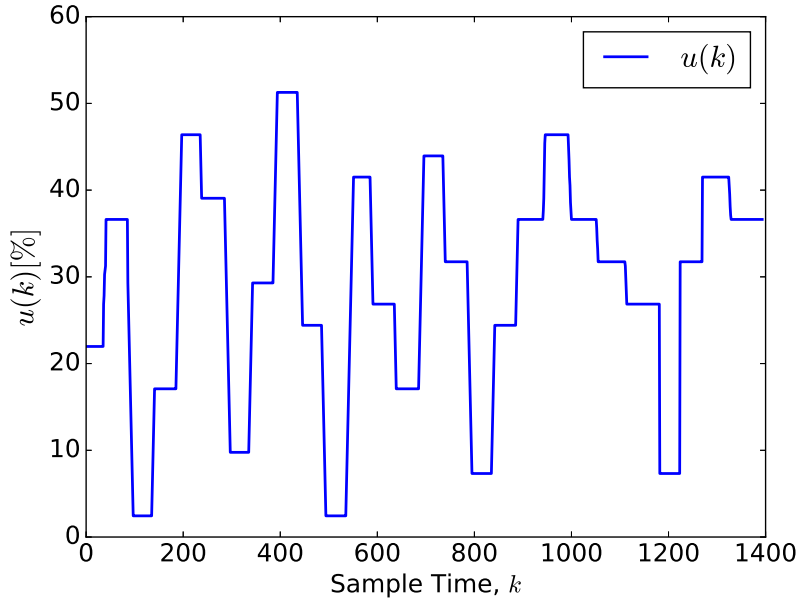
ator is connected to the "variable load" $R_L(k)$ that is also controlled using PWM signals. Managing the plant, there is a Texas Instruments Tiva C micro-controller (μC). This μC establishes serial communication with a Computer where the control algorithms are running, receives through serial communication the PWM values to control both the PWM of the motor and the PWM of the load, delivers PWM signals both to the motor driver and the load, reads the motor's encoder and finally sends back to the computer the encoder's value using serial communication.

The major feature of this plant is its "variable load", which is achieved using an electronic circuit from Arachnid Labs named Re:load (Figure 4.9) with a simple modification to allow it to be controlled by PWM. To be able to control the load from the computer is was needed to remove the potentiometer from Re:load and deliver an analog voltage to the circuit. Because the used μC only has digital and PWM outputs it was needed to build a RC filter and a voltage divider (Figure 4.8(b)) to be able to deliver a continuous voltage to the circuit' op-amp using a PWM output. This plant allows the control/command of both the motor and of the "heaviness" of the load by sending integer commands in the range from 0 to 4096. The sampling period in this real-world experiment is, $T_s = 100\,[\text{ms}]$.

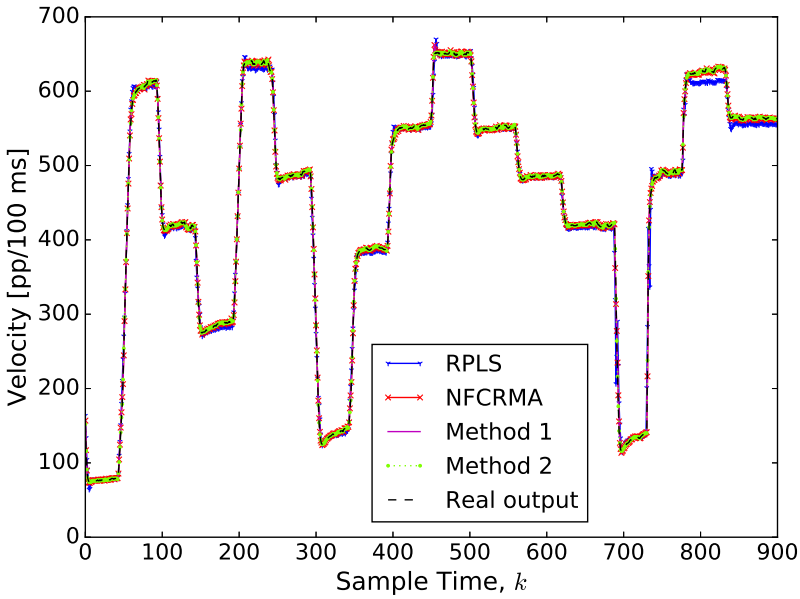
Identification of the Two Coupled DC Motors Process

To identify the experimental setup, a data set was constructed. The data set was obtained by applying to the motor the control signal represented in Figure 4.10(a).

The variables chosen for the data set were the first five delayed versions of the velocity $[y(k-1),\, y(k-2),\, y(k-3),\, y(k-4),\, y(k-5)]$, and the command signal, $u(k)$, and its first five delayed versions $[u(k-1),\, u(k-2),\, u(k-3),\, u(k-4),\, u(k-5)]$, where k is the sample time. For the DC motors application, the available data set was split into 35% for training, and the remaining 65% was used as the test data set. In NUFCA, the best number of clusters was determined as $c^* = 9$. The degree of fuzziness was chosen as $\eta = 2$. Numerical results comparing the performance of Method 1 and Method 2, and the works RPLS [Dayal and MacGregor, 1997], MLPNN [Nelles, 2001], NFCRMA [Li *et al.*, 2009], APSO [Alfi and Modares, 2011], ILLSA [Kadlec and Gabrys, 2011], ELM [Huang *et al.*, 2012], and HGA [Mendes *et al.*, 2012] are presented in Table 4.5. Method 2 produced 26 fuzzy rules, and the inputs of the T-S fuzzy system model learned (selected) by the HPSO (Algorithm 4.2) are $[y(k-1),\, y(k-3),\, u(k-1),\, u(k-2),\, u(k-4)]$. The same inputs were chosen in Method 1 (Algorithm 4.1). Figure 4.10(b) shows the comparison of the velocity values of the motor obtained by Method 1, Method 2, and by the RPLS and NFCRMA methods which were randomly chosen to be plotted, and the real/observed velocity values. From the results of Figure 4.10(b), and Table 4.4, it can be seen that the modeling of the velocity by the proposed methodologies, i.e. Method 1 and Method 2, is accurate and has a better prediction performance



(a)



(b)

Figure 4.10: Motor data set: (a) control signal used to compile the data set on the two coopled DC motors process, and (b) modelling performance of the proposed Algorithm 4.1 (which uses RLS-ADF method for online adaptive modeling), Algorithm 4.2 (which uses Algorithm 4.3 for online adaptive modeling), and comparing with RPLS [Dayal and MacGregor, 1997] and NFCRMA [Li *et al.*, 2009].

Table 4.5: Comparison of the system identification results on the test data set for the two coupled DC motors setup.

Method	Number of rules	Number of inputs	Inputs	1/MSE
RPLS [Dayal and MacGregor, 1997]	-	-	All variables	0.1634
MLPNN [Nelles, 2001]	-	-	All variables	0.1907
NFCRMA [Li *et al.*, 2009]	-	-	All variables	0.1850
FCM-APSO [Alfi and Modares, 2011]	-	-	All variables	0.2274
ILLSA [Kadlec and Gabrys, 2011]	-	-	All variables	0.1122
ELM [Huang *et al.*, 2012]	-	-	All variables	0.1771
HGA [Mendes *et al.*, 2012]	20	3	$y(t-2)$, $y(t-4)$, $u(t-2)$	0.158
Method 1	19	5	$y(k-1)$, $y(k-3)$, $u(k-1)$, $u(k-2)$, $u(k-4)$	0.3441
Method 2	26	5	$y(k-1)$, $y(k-3)$, $u(k-1)$, $u(k-2)$, $u(k-4)$	0.4682

when compared to the modeling obtained by the RPLS and NFCRMA methods. Also, the estimated T-S fuzzy model given by Method 2 is more accurate to model the real/observed velocity values when comparing to the estimated T-S fuzzy model given by Method 1. However, the implementation of Method 2 uses the results of the NUFCA performance which is a part of Method 1.

4.5 Conclusion

Two adaptive methodologies for online learning of Takagi-Sugeno (T-S) fuzzy models based on two different techniques were proposed in this chapter.

First, a new unsupervised fuzzy clustering algorithm (NUFCA) was proposed to construct a novel online evolving Takagi-Sugeno (T-S) fuzzy model identification method. The proposed system identification approach consists of two main steps: antecedent T-S fuzzy model parameters identification and consequent parameters identification. The NUFCA combines the K-nearest neighbour and fuzzy C-means methods into a fuzzy modelling method for partitioning of the input data and identifying the antecedent parameters of the fuzzy system; then the RLS-ADF method is exploited to obtain initial consequent parameters and to construct a method for on-line fuzzy model identification.

Second, a T-S fuzzy model identification methodology using a hierarchical fuzzy

structure was proposed. A novel learning algorithm based on a Hierarchical Particle Swarm Optimization (HPSO) was introduced to automatically extract all the fuzzy logic system (FLS)'s parameters of a T-S fuzzy model. The HPSO was proposed to automatically extract all the parameters of the T-S fuzzy model from a set of input/output data. Furthermore, a self-adaptive HPSO (S-AHPSO) algorithm was proposed for online identification of T-S fuzzy models.

To validate and demonstrate the performance and effectiveness of the proposed algorithms, the identification of a real WWTP plant, the identification of the dissolved oxygen in a simulated activated sludge reactor within a WWTP, and the identification of a real-world experimental setup composed of two coupled DC motors were studied. The results show that the proposed techniques, in both methodologies, can successfully identify fuzzy model parameters that represent the dynamics of nonlinear plants, using only a data set of the process, where the model can be further used to estimate the output of the plant. Moreover, the results reveal superior performance of the proposed identification methods when compared to other state of the art methods. In the next chapter, the fuzzy modeling methodologies proposed in this chapter will be used to design adaptive fuzzy generalized predictive controllers, a proposed solution to address the concerns in the control design for nonlinear systems, and in particular with the presentation of experiments in the control of a simulated WWTP plant, and in the velocity control of the real-world experimental setup composed of two coupled DC motors.

Chapter 5

Adaptive Fuzzy Model Predictive Control

Contents

5.1 Introduction / State of the Art

Model predictive control (MPC) is a popular control approach which is widely used in practice due to its high-quality control performance. The main objective in model predictive control (MPC) design is to find a future trajectory of the manipulated

input variable (command) in such a way that it can optimize the future behavior of the plant output. The optimization procedure is performed in a finite time window by giving the current plant status at the start of the time window. The MPC are designed based on a mathematical model of the plant. There are three well-known model structures which can be seen frequently in MPC designs. The MPC control designs which use finite impulse response (FIR) models as well as step response models, MPC designs based on state space models, and MPC designs based on transfer function models. Among all MPC designs which use transfer function models, maybe the most typical controller is the generalized predictive control (GPC) [Camacho and Bordons, 2007]. The GPC is applicable to both stable and unstable plants, and has shown good performance results [Clarke, 1988; Tham *et al.*, 1991] using linear plant models. However, a main drawback of GPC, as commonly in MPCs, is its assumption of the knowledge of an accurate model of the process to be controlled. Frequently, in industrial environments, physical systems contain complex nonlinear behaviors and relations. In most cases, they are difficult to be modeled with conventional techniques.

As already mentioned in this thesis, the assumption of the knowledge of an accurate model in MPCs, presents problems because many complex plants are difficult to be mathematically modeled based on physical laws, or have large uncertainties and strong nonlinearities. Among several modeling approaches which have been applied, fuzzy logic systems have received particular attention in the area of nonlinear modeling, which is theoretically supported by the fact that fuzzy logic systems are universal approximators [Wang and Mendel, 1992; Kosko, 1994].

[Smoczek, 2015] tried to address the issue of reducing the residual vibration and limiting the transient oscillations of a flexible and under-actuated system with respect to the variation of operating conditions. Two alternatives of a GPC-based method were proposed that enable to realize this technique either with or without a sensor of payload deflection. The first control technique was based on the recursive least squares (RLS) method applied to on-line estimate the parameters of a linear parameter varying (LPV) model of a crane dynamic system. The second GPC-based approach was based on a payload deflection feedback estimated using a pendulum model with the parameters interpolated using the T-S fuzzy system. In [Wu *et al.*, 2012] a GPC strategy with closed-loop model identification for burn-through point (BTP) control in the sintering process, was proposed. First, the dynamic Auto-

Regressive eXogenous (ARX) model structure was defined to describe the sintering process. Then, BTP predictive control model was established based on the GPC algorithm to predict BTP accurately and to calculate the strand velocity. In [Bello *et al.*, 2014], a fuzzy model predictive control strategy was proposed to regulate the output variables of a coagulation chemical dosing unit. A multiple-input, multiple-output (MIMO) process model in the form of a linearised Takagi-Sugeno (T-S) fuzzy model was derived. The process model was obtained from the plant's data set through subtractive clustering. Killian *et al.* [2014] presented a hierachical fuzzy model predictive control structure with decoupled MPCs for building heating control using weather forecasts and occupancy information. The hierachical MPC in [Killian *et al.*, 2014] could only find local optima. Therefore, a sub-optimal solution was presented to solve this problem. Indeed, both fuzzy MPC designs in [Bello *et al.*, 2014; Killian *et al.*, 2014] were designed to control nonlinear plants. But, none of the works in [Bello *et al.*, 2014; Killian *et al.*, 2014] proposed any adaptive strategy in T-S fuzzy modelling to update parameters in the plants modelling. Some recent works are available which can be found interesting by fuzzy MPC designers. For example in the work [Mendes, 2014], issues such as online fuzzy MPC design, direct fuzzy control, as well as different techniques to construct the T-S fuzzy model have been presented and discussed. The work followed computational intelligence methodologies to learn fuzzy systems from data, and then combined the learned fuzzy models with GPC for control purposes in industrial processes. Using different hierachical fuzzy structures usable in both T-S fuzzy system modelling and fuzzy control designs are significant achievements of this work. In [Han *et al.*, 2012] a self-organizing radial basis function neural network MPC method is proposed for controlling the dissolved oxygen concentration in a WWTP. The methods of [Kayadelen, 2011; Hung and Lin, 2012; Han *et al.*, 2012; Wu *et al.*, 2012; Smoczek, 2015], have the limitation of not being able to perform automatic selection of variables and delays: pre-selection is performed. The variable selection process is usually manual and not accompanied with the accurate selection of the right time delays, probably leading to low-accuracy results. As previously mentioned in this thesis, a variable with the correct delay may contain more information about the output, than one which does not consider any delay or which considers an incorrect delay [Souza *et al.*, 2010].

This chapter is going to deal with the problem of the assumption of knowledge about an accurate model of the process in the MPC control framework. For this

purpose, the identification methodologies proposed in Chapter 4 are combined with the GPC algorithm. The integration of the T-S fuzzy models learned in Chapter 4, and GPC result in effective adaptive fuzzy GPC methodologies (AFGPC) which are applied for the control of nonlinear processes.

5.2 Adaptive Fuzzy Model Predictive Control Design

As discussed in Section 3.3, adaptive fuzzy control designs can be categorized into two different types: direct adaptive fuzzy control (DAFC), and indirect adaptive fuzzy control (IAFC). The controllers proposed in this chapter can be categorized into the second type, where a fuzzy logic modeling approach is integrated with a well-known GPC control structure to construct a fuzzy model predictive controller. Also, the adaptive fuzzy model identification approaches proposed in Chapter 4, will be used for online control purposes.

After having studied the identification algorithms in Section 4.2 (Algorithm 4.1) and Section 4.3 (Algorithms 4.2, and 4.3), in this section the control algorithms are presented. A diagram of the adaptive fuzzy generalized predictive control (AFGPC) approach is presented in Figure 5.1. As can be seen, the control scheme consists of the plant, the controller, and the adaptive T-S fuzzy model. The controller is composed of a model-based predictive controller that integrates a T-S fuzzy model. Two different T-S fuzzy model identification methodologies were introduced in Chapter 4. Correspondingly, two AFGPC schemes based on two different T-S fuzzy models are designed. The first AFGPC algorithm uses the model results extracted from Algorithm 4.1 to be constructed. The consequent parameters of the model are adjusted the RLS-ADF adaptation law studied in Subsection 4.2.5. The second proposed AFGPC integrates a T-S fuzzy model learned off-line, according to the methodology presented on Algorithm 4.2. In the second AFGPC, in online mode, the consequent parameters of the model are also adjusted in a recursive procedure using the adaptation procedure given by Algorithm 4.3.

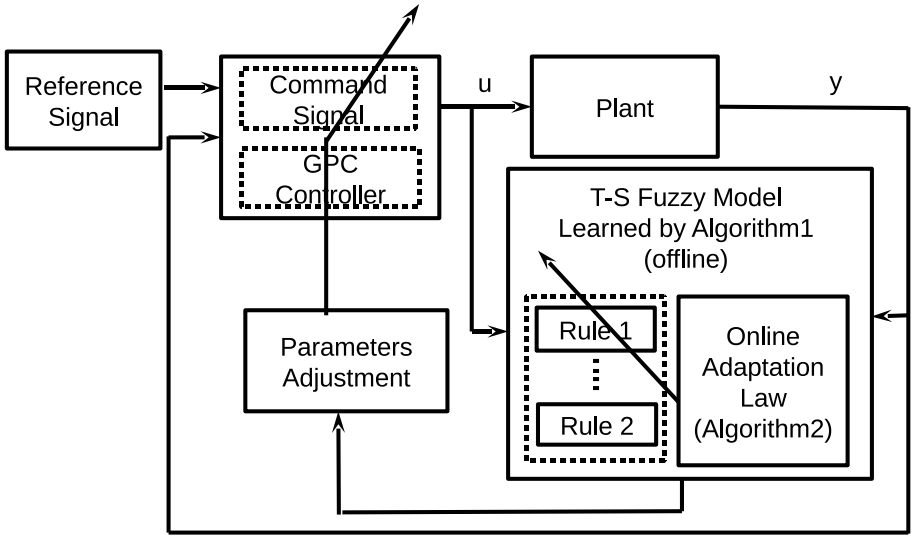


Figure 5.1: A generic schematic diagram of the AFGPC control architecture.

5.2.1 Dynamic Systems Modelling Using T-S Fuzzy Models

Dynamic systems of practical interest often have significant nonlinearities. Processes of the following form will be considered [Rastegar *et al.*, 2016b; Mendes *et al.*, 2013]:

$$y(k) = f[y(k-1), \ldots, y(k-n_y), u(k-\bar{d}-1), \ldots, u(k-\bar{d}-n_u)], \tag{5.1}$$

where $f(\cdot) : \mathbb{R}^{n_y+n_u} \to \mathbb{R}$ is a nonlinear function which is assumed to be unknown, $y(\cdot) : \mathbb{N} \to \mathbb{R}$ is the system output, $u(\cdot) : \mathbb{N} \to \mathbb{R}$ is the system input, $\bar{d} \in \mathbb{N}$ is the dead time, $\bar{d}+1$ is the time delay of the system, and $n_u \in \mathbb{N}$ and $n_y \in \mathbb{N}$ are the orders of the input and output, respectively. A large class of nonlinear SISO processes can be represented by (5.1). $f(\cdot)$ is approximated by a T-S fuzzy system. For the GPC controller, system (5.1) can be described by a T-S fuzzy model defined by the following fuzzy rules:

$$\begin{aligned} R_i: \ & \text{IF } x_1(k) \text{ is } A_{i1}, \text{ and } \ldots \text{ and } x_N(k) \text{ is } A_{iN} \\ & \text{THEN } y_i(k) = a_i(z^{-1})y(k-1) + b_i(z^{-1})u(k-\bar{d}-1), \quad i = 1, \ldots, c, \end{aligned} \tag{5.2}$$

where $u(k)$ is the command signal, c is the number of rules, and $N = n_y + n_u$.

The polynomials $a_i(z^{-1})$, and $b_i(z^{-1})$ in (5.2) are defined as follows:

$$\begin{aligned} a_i(z^{-1}) &= a_{1i} + a_{2i}z^{-1} + \ldots + a_{n_y i}z^{-(n_y-1)}, \\ b_i(z^{-1}) &= b_{1i} + b_{2i}z^{-1} + \ldots + b_{n_u i}z^{-(n_u-1)}, \end{aligned} \tag{5.3}$$

and $\mathbf{x}(k) = [x_1(k), \ldots, x_N(k)] = [y(k-1), \ldots, y(k-n_y), u(k-\bar{d}-1), \ldots, u(k-\bar{d}-n_u)]$ is the vector of input variables of the T-S fuzzy system. Thus, from (5.2) $y(k)$ can be rewritten as

$$\begin{aligned} y(k) &= \sum_{i=1}^{c} \bar{\omega}^i[\mathbf{x}(k)] \left[a_i(z^{-1})y(k-1) + b_i(z^{-1})u(k-\bar{d}-1)\right], \qquad (5.4) \\ &= \sum_{i=1}^{c} \bar{\omega}^i[\mathbf{x}(k)]\mathbf{x}(k)\boldsymbol{\theta}_i, \\ &= \boldsymbol{\Psi}(k)\boldsymbol{\Theta}, \qquad (5.5) \end{aligned}$$

where for $i = 1, \ldots, c$,

$$\begin{aligned} \bar{\omega}^i[\mathbf{x}(k)] &= \frac{\prod_{j=1}^{N} A_{ij}^h(x_j)}{\sum_{i=1}^{c}\prod_{j=1}^{N} A_{ij}^h(x_j)}, \qquad (5.6) \\ \boldsymbol{\theta}_i &= \left[a_{1i}, \ldots, a_{n_y i}, b_{1i}, \ldots, b_{n_u i}\right]^T, \qquad (5.7) \\ \boldsymbol{\Theta} &= \left[\boldsymbol{\theta}_1^T, \boldsymbol{\theta}_2^T, \ldots, \boldsymbol{\theta}_c^T\right]^T, \qquad (5.8) \\ \boldsymbol{\Psi}(k) &= \left[(\bar{\omega}_1[\mathbf{x}(k)])\,\mathbf{x}(k), \ldots, (\bar{\omega}_c[\mathbf{x}(k)])\,\mathbf{x}(k)\right]. \qquad (5.9) \end{aligned}$$

5.2.2 Generalized Predictive Control Law

It is assumed that the plant model is of the form (5.5), which can be rewritten as follows [Rastegar *et al.*, 2016b; Mendes *et al.*, 2013]:

$$\bar{a}(z^{-1})y(k) = \bar{b}(z^{-1})u(k-\bar{d}-1), \tag{5.10}$$

where

$$\begin{aligned} \bar{a}(z^{-1}) &= 1 - \bar{a}_1 z^{-1} - \ldots - \bar{a}_{n_y} z^{-n_y}, \qquad (5.11) \\ \bar{b}(z^{-1}) &= \bar{b}_1 + \bar{b}_2 z^{-1} + \ldots + \bar{b}_{n_u} z^{-(n_u-1)}, \qquad (5.12) \end{aligned}$$

$$\bar{a}_t = \sum_{i=1}^{c} \bar{\omega}^i[\mathbf{x}(k)]a_{ti}, \;\; t = 1, \ldots, n_y, \tag{5.13}$$

$$\bar{b}_m = \sum_{i=1}^{c} \bar{\omega}^i[\mathbf{x}(k)]b_{mi}, \;\; m = 1, \ldots, n_u. \tag{5.14}$$

The GPC control law is obtained so as to minimize the following cost function

$$J(k) = \sum_{p=\bar{d}+1}^{N_p} \left[\hat{y}(k+p|k) - r(k+p)\right]^2 + \sum_{p=\bar{d}+1}^{\bar{d}+N_u} \left[\lambda(z^{-1})\Delta u(k+p-\bar{d}-1|k)\right]^2, \tag{5.15}$$

where $\hat{y}(k+p|k)$ is a p-step ahead prediction of the system on instant k, $r(k+p)$ is the future reference trajectory, $\Delta = 1 - z^{-1}$, and $\lambda(z^{-1}) = \lambda_0 + \lambda_1 z^{-1} + \ldots + \lambda_{N_p+n_u-1} z^{-(N_p+n_u-1)}$ is a weighting polynomial. N_p and N_u are the output and control horizons, respectively. Consider the following Diophantine equation (5.16):

$$1 = \Delta e_p(z^{-1})\bar{a}(z^{-1}) + z^{-p} f_p(z^{-1}), \tag{5.16}$$

In (5.16) the two polynomials $e_p(z^{-1})$, and $f_p(z^{-1})$ are defined as follows:

$$e_p(z^{-1}) = 1 + e_{p,1}z^{-1} + \ldots + e_{p,p-1}z^{-(p-1)}, \tag{5.17}$$

$$f_p(z^{-1}) = f_{p,0} + f_{p,1}z^{-1} + \ldots + f_{p,n_y}z^{-n_y}, \tag{5.18}$$

where $e_p(z^{-1})$ and $f_p(z^{-1})$ can be obtained by dividing 1 by $\Delta\bar{a}(z^{-1})$ until the remainder can be factorized as $z^{-p}f_p(z^{-1})$. The quotient of the division is the polynomial $e_p(z^{-1})$. A simple and efficient way to obtain polynomials $e_p(z^{-1})$ and $f_p(z^{-1})$ is to use recursion of the Diophantine equation as demonstrated in [Camacho and Bordons, 2007]. Polynomials $e_{p+1}(z^{-1})$ and $f_{p+1}(z^{-1})$ can be obtained from polynomials $e_p(z^{-1})$ and $f_p(z^{-1})$, respectively. Polynomials $e_{p+1}(z^{-1})$ are given by

$$e_{p+1}(z^{-1}) = e_p(z^{-1}) + z^{-p}e_{p+1,p}, \tag{5.19}$$

where $e_{p+1,p} = f_{p,0}$. The coefficients of polynomial $f_{p+1}(z^{-1})$ can be obtained recursively as follows:

$$f_{p+1,i} = f_{p,i+1} - f_{p,0}\,\Delta\bar{a}_{i+1}, \quad i = 0, \ldots, n_y - 1, \tag{5.20}$$

where $f_{p,n_y} = 0$. Polynomial $g_{p+1}(z^{-1})$ is expressed as:

$$\begin{aligned} g_{p+1}(z^{-1}) &= e_{p+1}(z^{-1})\bar{b}(z^{-1}) = \left[e_p(z^{-1}) + z^{-p}f_{p,0}\right]\bar{b}(z^{-1}) \\ &= g_p(z^{-1}) + z^{-p}f_{p,0}\bar{b}(z^{-1}), \end{aligned} \tag{5.21}$$

where the coefficients of $g_{p+1}(z^{-1})$ are given by $g_{p+1,j} = g_{p,j}$ for $j = 0, \ldots, p-1$, and

$$g_{p+1,p+i} = g_{p,p+i} + f_{p,0}\bar{b}_i, \quad i = 0, \ldots, n_u, \tag{5.22}$$

where $g_{p,p+n_u} = 0$. $e_p(z^{-1})$, $f_p(z^{-1})$, and $g_p(z^{-1})$ are recursively computed for $p = \bar{d}+1, \ldots, N_p$. To initialize the recursion (5.16), $p = \bar{d}+1$, and

$$e_{\bar{d}+1}(z^{-1}) = 1, \tag{5.23}$$

$$f_{\bar{d}+1}(z^{-1}) = z(1 - \tilde{a}(z^{-1})) = \tilde{a}_1 + \tilde{a}_2 z^{-1} + \ldots + \tilde{a}_{n_y+1}z^{-n_y}, \tag{5.24}$$

where

$$\tilde{a}(z^{-1}) = \Delta\bar{a}(z^{-1}) = 1 - \tilde{a}_1 z^{-1} - \ldots - \tilde{a}_{n_y+1}z^{-(n_y+1)}. \tag{5.25}$$

Replacing (5.23) into (5.21), and considering to $p = \bar{d}$ gives

$$g_{\bar{d}+1}(z^{-1}) = e_{\bar{d}+1}(z^{-1})\bar{b}(z^{-1}) = \bar{b}(z^{-1}). \tag{5.26}$$

Multiplying (5.10) by $\Delta z^p e_p(z^{-1})$ yields

$$\Delta z^p e_p(z^{-1})\bar{a}(z^{-1})y(k) = \Delta z^p e_p(z^{-1})\bar{b}(z^{-1})u(k - \bar{d} - 1). \tag{5.27}$$

Defining

$$\begin{aligned} g_p(z^{-1}) &= e_p(z^{-1})\bar{b}(z^{-1}), \\ &= g_{p,0} + g_{p,1}z^{-1} + \ldots + g_{p,p+n_u-1}z^{-(p+n_u-1)}, \end{aligned} \tag{5.28}$$

and substituting (5.16) and (5.28) into (5.27) gives

$$y(k+p|k) = f_p(z^{-1})y(k) + g_p(z^{-1})\Delta u(k + p - \bar{d} - 1). \tag{5.29}$$

Thus, the best prediction of $y(k+p|k)$ is

$$\hat{y}(k+p|k) = f_p(z^{-1})y(k) + g_p(z^{-1})\Delta u(k+p-\bar{d}-1). \tag{5.30}$$

Equation (5.30) can be rewritten as

$$\mathbf{y}(k) = \mathbf{G}\mathbf{u}(k) + \mathbf{F}(z^{-1})y(k) + \mathbf{L}(z^{-1}), \tag{5.31}$$

where

$$\mathbf{y}(k) = \begin{bmatrix} \hat{y}(k+\bar{d}+1) \\ \hat{y}(k+\bar{d}+2) \\ \vdots \\ \hat{y}(k+N_p) \end{bmatrix}, \mathbf{u}(k) = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_u-1) \end{bmatrix}, \tag{5.32}$$

$$\mathbf{F} = \begin{bmatrix} f_{\bar{d}+1}(z^{-1}) \\ f_{\bar{d}+2}(z^{-1}) \\ \vdots \\ f_{N_p}(z^{-1}) \end{bmatrix}, \mathbf{G} = \begin{bmatrix} g_{1,0} & 0 & \dots & 0 \\ g_{2,1} & g_{2,0} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_p,N_p-1} & g_{N_p,N_p-2} & \dots & g_{N_p,N_p-N_u} \end{bmatrix}, \tag{5.33}$$

$$\mathbf{L} = \begin{bmatrix} \left[g_{\bar{d}+1}(z^{-1}) - \bar{g}_{\bar{d}+1}(z^{-1})\right] z\Delta u(k-1) \\ \left[g_{\bar{d}+2}(z^{-1}) - \bar{g}_{\bar{d}+2}(z^{-1})\right] z^2\Delta u(k-1) \\ \vdots \\ \left[g_{N_p}(z^{-1}) - \bar{g}_{N_p}(z^{-1})\right] z^{N_p}\Delta u(k-1) \end{bmatrix},$$

$$\bar{g}_p(z^{-1}) = g_{p,0} + g_{p,1}z^{-1} + \ldots + g_{p,p-\bar{d}-1}z^{\bar{d}+1-p}.$$

Using (5.31) and considering $\lambda(z^{-1})$ to be constant ($\lambda > 0$), (5.15) can be rewritten as

$$J_{eq}(k) = \left[\mathbf{F}y(k) + \mathbf{G}\mathbf{u}(k) + \mathbf{L} - \mathbf{R}\right]^T \left[\mathbf{F}y(k) + \mathbf{G}\mathbf{u}(k) + \mathbf{L} - \mathbf{R}\right] + \left[\lambda\mathbf{u}(k)\right]^2, \tag{5.34}$$

where

$$\mathbf{R} = \left[r(k+\bar{d}+1), \ldots, r(k+N_p)\right]^T. \tag{5.35}$$

The necessary condition of the minimum $J_{eq}(k)$ is obtained through the first deriva-

tive of cost funtion $J_{eq}(k)$, and by

$$\frac{\partial J_{eq}(k)}{\partial[\Delta u(k)]} = 0. \tag{5.36}$$

from which the optimal solution for the control signal is found as [Camacho and Bordons, 2007]

$$\mathbf{u}^*(k) = \frac{\mathbf{G}^T(\mathbf{R} - \mathbf{F}y(k) - \mathbf{L})}{\mathbf{G}^T\mathbf{G} + \lambda\mathbf{I}}, \tag{5.37}$$

where $\mathbf{I}$ is the identity matrix. As the control signal sent to the process is the first row of $\mathbf{u}^*(k)$, the $\Delta u^*(k)$ is given by:

$$\Delta u^*(k) = \mathbf{K}[\mathbf{R} - \mathbf{F}y(k) - \mathbf{L}], \tag{5.38}$$

where $\mathbf{K}$ is the first row of matrix $(\mathbf{G}^T\mathbf{G} + \lambda\mathbf{I})^{-1}\mathbf{G}^T$,

$$\mathbf{K} = \begin{bmatrix}1 & 0 & 0 & \cdots & 0\end{bmatrix}_{1\times N_u}(\mathbf{G}^T\mathbf{G} + \lambda\mathbf{I})^{-1}\mathbf{G}^T. \tag{5.39}$$

5.2.3 Fuzzy Predictive Control Scheme Algorithms

The main steps of the two proposed fuzzy model predictive control architectures are presented in this section. The two fuzzy predictive control frameworks are proposed in Algorithm 5.1 and Algorithm 5.2:

1. Algorithm 5.1 summarizes the design and operation of the first adaptive fuzzy generalized predictive controller [Rastegar *et al.*, 2016b]. The fuzzy controller uses the fuzzy model results which are extracted by Algorithm 4.1, where the consequent parameters of the model in this fuzzy controller are adjusted in a recursive procedure using the RLS-ADF adaptation law studied in Subsection 4.2.5.

2. Algorithm 5.2 proposes the second type of adaptive fuzzy model predictive controller for this section. This second fuzzy controller uses the T-S fuzzy model learned by Algorithm 4.2, while the consequent parameters of the model in this fuzzy controller are adjusted in a recursive adaptive procedure given by Algorithm 4.3 (S-AHPSO).

Algorithm 5.1 Adaptive fuzzy generalized predictive control algorithm based on the T-S fuzzy modelling system proposed in Algorithm 4.1, and the RLS-ADF methodology (Subsection 4.2.5).

(a) Design control parameters: N_p, N_u, λ and $\bar{d}$. Design the identification parameters (ρ, φ_i, τ_i, ν_i, for $i = 1, \ldots, c$) with the same values as the ones defined in Algorithm 4.1;

(b) Use the fuzzy rule base (input variables, respective membership functions, the fuzzy rules and the final learned model parameters) learned in Algorithm 4.1 and initialize $u(0)$;

(c) For/using each newly arriving online sample, do:

i. Compute $\bar{a}(z^{-1})$ and $\bar{b}(z^{-1})$ using (5.11) and (5.12), respectively;

ii. Compute control signal $\Delta u(k)$ with (5.38);

iii. Adapt the T-S fuzzy model parameters (a_{ji} and b_{ji} of (5.3)) by performing one iteration of recursion (4.16).

Algorithm 5.2 Adaptive fuzzy generalized predictive control algorithm based on the hierarchical T-S fuzzy modelling system proposed in Algorithm 4.2, and the S-AHPSO methodology proposed in Algorithm 4.3.

(a) Use the fuzzy rule base (input variables, respective membership functions, the fuzzy rules and the final learned model parameters) learned in Algorithm 4.2 and initialize $u(0)$;

(b) Design control parameters: N_p, N_u, λ and $\bar{d}$. Design the identification parameters ($w_{r,max}$, $w_{r,min}$, c_{1r}, c_{2r}, r_{max}, v_{min}, v_{max}, γ, β, and T_2) with the same values as the ones defined in Algorithm 4.3;

(c) For/using each newly arriving online sample, do:

i. Compute $\bar{a}(z^{-1})$ and $\bar{b}(z^{-1})$ using (5.11) and (5.12), respectively;

ii. Compute control signal $\Delta u(k)$ with (5.38);

iii. Adapt the T-S fuzzy model parameters (a_{ji} and b_{ji} of (5.3)) by performing one iteration of recursion (4.28).

5.3 Experimental Results

This section presents simulation and real-world results to demonstrate the effectiveness of both proposed adaptive fuzzy model predictive control methods proposed in Section 5.2. The performance of the proposed adaptive predictive fuzzy control methodologies are demonstrated on two setups: a simulated WWTP plant (Benchmark Simulation Model 1 (BSM1), Subsection 4.4.2), and a real-world experimental setup composed of two coupled DC motors (Subsection 4.4.4). For the WWTP application, a comparison between the performance of the two AFGPC proposed in this work with the PID controller, a direct fuzzy controller (DFC) design [Mendes *et al.*, 2014], and the classical GPC [Camacho and Bordons, 2007], is shown and discussed. For the two DC motors real-world setup, a comparison between the performance of the two AFGPC proposed in this work with the classical GPC [Camacho and Bordons, 2007], is presented and discussed. In all the simulations, the notation "AFGPC (Method 1)" refers to the AFGPC scheme of Algorithm 5.1, which is based on the T-S fuzzy model obtained by Algorithm 4.1, while the AFGPC in this design uses RLS-ADF (Subsection 4.2.5) to update the consequent parameters of the constructed T-S fuzzy model. Also, the notation "AFGPC (Method 2)" refers to the AFGPC scheme of Algorithm 5.2, which is based on the T-S fuzzy model learned by Algorithm 4.2 (HPSO), and the adaptation procedure given by Algorithm 4.3 (S-AHPSO).

5.3.1 Adaptive Predictive Fuzzy Control of the Simulated BSM1 Plant

The two adaptive fuzzy model predictive control methods proposed in Section 5.2, using the two different T-S fuzzy modelling methods presented in Section 4.2, and Section 4.3 are tested on the BSM1 plant and compared with other controllers. For the first type of AFGPC, the model learned by Algorithm 4.1 and being updated by RLS-ADF (i.e. the model is learned by Method 1, Subsection 4.2.5) is used to initialize the prediction model of the adaptive fuzzy GPC controller of Subsection 5.2.3. For the second type of AFGPC, the model learned by Algorithm 4.2 and updated by Algorithm 4.3 (Subsection 4.3.3) is used to initialize the prediction model of the adaptive fuzzy GPC controller of Subsection 5.2.3. Also, in the online control

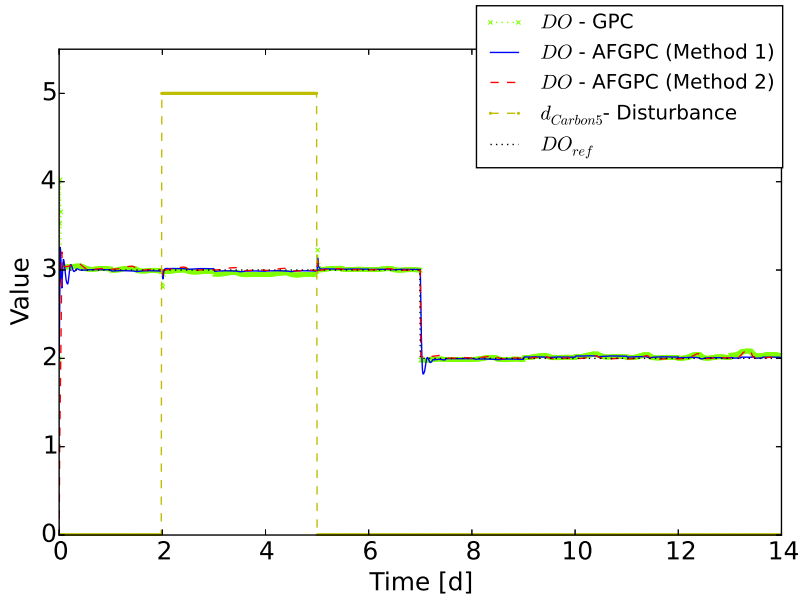
operation, the RLS-ADF method and the Algorithm 4.3 are used to update the consequent parameters of the first type and second type of AFGPC, respectively. For the AFGPC which uses Method 1, the following controller parameters were chosen: $N_p = 10$, $N_u = 1$, $\lambda = 0.08$, $\bar{d} = 0$, $\rho = 0.999$, $\varphi_i = 1$, $\tau_i = \nu_i = 1 \times 10^{-8}$, for $i = 1, \ldots, c$. For the AFGPC which uses Method 2, the following controller parameters were: $N_p = 5$, $N_u = 1$, $\lambda = 8 \times 10^{-5}$, $\bar{d} = 0$, $w_{r,max} = 1$, $w_{r,min} = 0.1$, $c_{1r} = 1.5$, $c_{2r} = 2$, $r_{max} = 16$, $v_{max} = 1.8$, $v_{min} = -0.9$, $\gamma = 1$, $\beta = 0.2$, and $T_2 = 30$. The best control parameters for both AFGPC methodologies were found experimentally by trial and error, and considering the results obtained in Subsection 4.4.3.

The reference input $DO_{ref}(t)\,[\text{g.cm}^{-3}]$ is

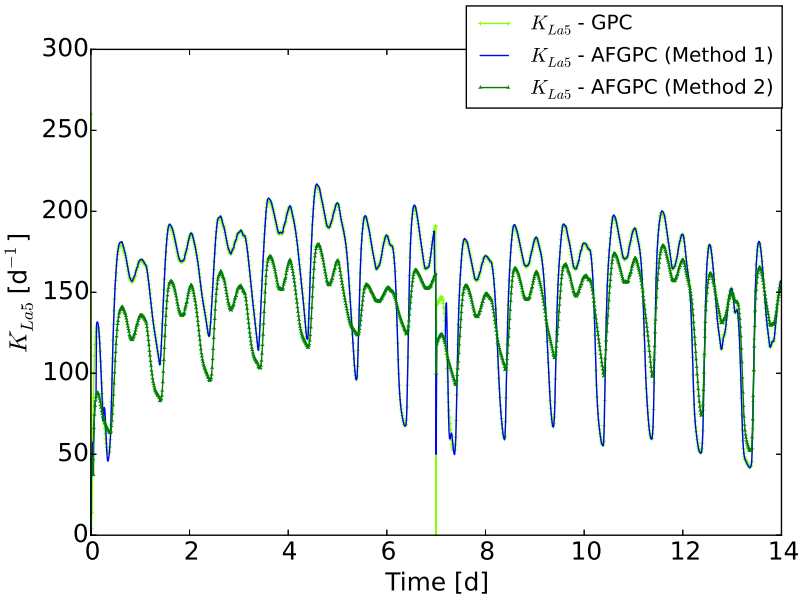
$$DO_{ref}(t) = \begin{cases} 3, & 0 < t \leqslant 7\,[\text{days}], \\ 2, & 7\,[\text{days}] < t \leqslant 14\,[\text{days}]. \end{cases} \tag{5.40}$$

As a source of disturbance, the external carbon (EC) source to reactor 5 is a constant equal to $d_{Carbon5}(k) = 0\,[\text{kg COD.days}^{-1}]$ (COD is the abbreviation for chemical oxygen demand), for $0 \leq t \leq 14\,[\text{days}]$, except that it is changed to a disturbance value of $d_{Carbon5}(k) = 5\,[\text{kg COD.days}^{-1}]$ during $2 \leq t \leq 5\,[\text{days}]$.

From the results presented in Figure 5.2(a), it can be seen that the proposed AFGPC methods which use the T-S fuzzy model identification methodologies proposed in Chapter 4 are able to adequately (attain and) control the system output at the desired reference $DO_{ref}(t)$. Numerical comparison of the results of the proposed adaptive fuzzy control methodologies with other controllers (PID, DFC, GPC) are reported in Table 5.1. For the PID controller, the best control parameters were found by the Ziegler-Nichols method [Ziegler and Nichols, 1942]. As shown in Table 5.1, comparing to the other methods (PID, DFC, GPC), with the proposed AFGPC (with two different T-S fuzzy models) more accuracy in tracking the reference $DO_{ref}(t)$ was obtained. From the results, it is concluded that the proposed AFGPC methodology can control the process using only a data set of the process to initialize the T-S fuzzy model. Comparing with the AFGPC (Method 1), a better performance can be seen by the AFGPC (Method 2) in the tracking of the set-points. The different control performances obtained by the AFGPC (Method 2) and AFGPC (Method 1) can be due to the important effect of each T-S fuzzy model



(a)



(b)

Figure 5.2: Dissolved oxygen control: (a) The results of the DO [g.m^{-3}] control with the proposed adaptive fuzzy predictive controllers of Algorithm 5.1, and Algorithm 5.2, and the classical GPC controller [Camacho and Bordons, 2007]; and (b) the respective applied K_{La5} command signals. The units of the disturbance variable $d_{Carbon5}(k)$ are [kg COD.days^{-1}].

Table 5.1: Comparison of results of the proposed adaptive fuzzy control methodologies, AFGPC, with other methods (PID, DFC, GPC), for the BSM1 test. Note that, here, the $MSE = \frac{1}{T_n}\sum_{t=1}^{T_n}(DO_{ref}(t) - DO(t))^2$ is related to the tracking error rather than to the estimation error that is used in the training of the T-S fuzzy models. T_n is the total number of samples on the test.

Methodology	$1/MSE = 1/\left(\frac{1}{T_n}\sum_{t=1}^{T_n}(DO_{ref}(t) - DO(t))^2\right)$
PID	335.3795
DFC	260.8385
GPC	382.8368
AFGPC (Method 1)	598.4009
AFGPC (Method 2)	774.7117

identification methodology, i.e. Method 1 and Method 2, in the control of the BSM1 plant. The fact that, except in the T-S fuzzy modeling part, both controllers use a similar control structure/procedure in the design, can indicate/confirm that a significant role is played by the T-S fuzzy model in a fuzzy control design. As was shown in the results of Subsection 4.4.3, Method 2 has shown a better estimation performance in T-S fuzzy model identification in the BSM1 plant, when compared to Method 1 (Table 4.2). The command signals obtained by the proposed controllers and by the other methods are depicted in Figure 5.2(b).

5.3.2 Adaptive Predictive Fuzzy Control of Two Coupled DC Motors

The two adaptive fuzzy model predictive controllers proposed in Section 5.2, and based on two different T-S fuzzy modelling methods (see Section 4.2, and Section 4.3), are tested on the real two coupled DC motors setup (Subsection 4.4.4). For the AFGPC which uses Method 1, the following controller parameters were chosen: $N_p = 8$, $N_u = 1$, $\lambda = 30$, $\bar{d} = 0$, $\rho = 0.95$, $\varphi_i = 1$, $\tau_i = 1 \times 10^{-3}, \nu_i = 1 \times 10^{-6}$, for $i = 1, \ldots, c$. For the AFGPC which uses Method 2, the following controller parameters were chosen by the user: $N_p = 10$, $N_u = 1$, $\lambda = 5 \times 10^{-3}$, $\bar{d} = 0$, $w_{r,max} = 0.9$, $w_{r,min} = 0.4$, $c_{1r} = 1.5$, $c_{2r} = 2$, $r_{max} = 20$, $v_{max} = 1.0$, $v_{min} = -0.8$, $\gamma = 1$, $\beta = 0.1$, and $T_2 = 20$. The best control parameters for both AFGPC methodologies were found experimentally by trial and error, and considering the results obtained in Subsection 4.4.4.

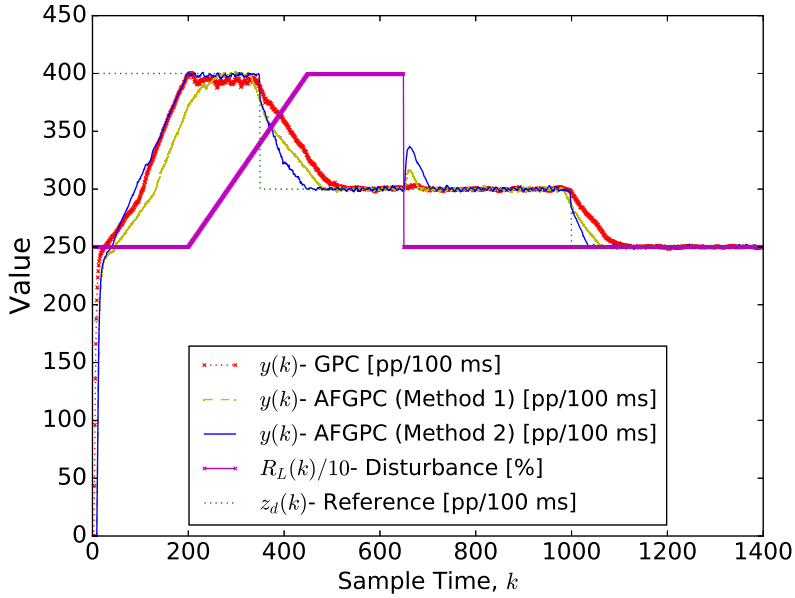
The reference input $z_d(t)$ [pp/(100 ms)] is

$$z_d(k) = \begin{cases} 400, & 0 < k \leqslant 350, \\ 300, & 350 < k \leqslant 1000, \\ 250, & 1000 < k \leqslant 1400, \end{cases} \tag{5.41}$$

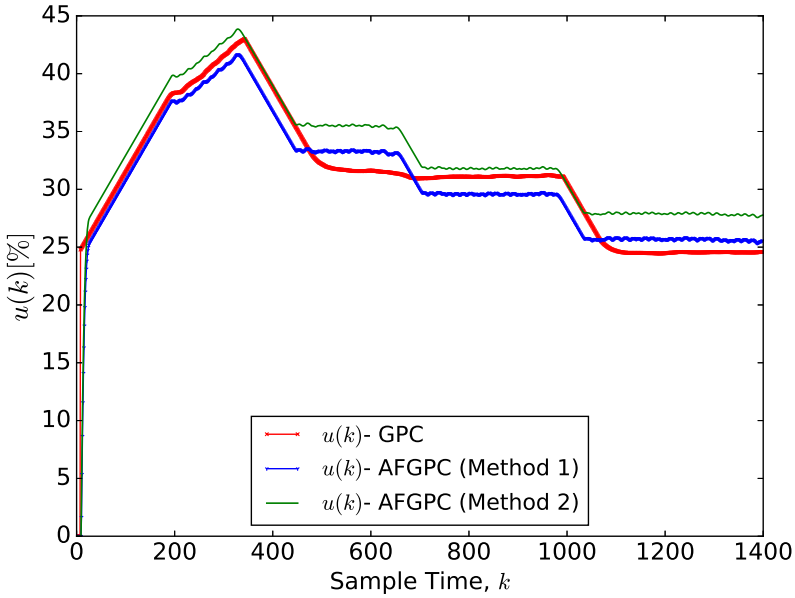
and the load disturbance $R_L(k)$ is:

$$R_L(k) = \begin{cases} 2500, & 0 < k \leqslant 200, \\ 6(k-200)+2500, & 200 < k \leqslant 450, \\ 4000, & 450 < k \leqslant 650, \\ 2500, & 650 < k \leqslant 1400. \end{cases} \tag{5.42}$$

$R_L(k)$ satisfies its admissible operating range which is $0 \leqslant R_L(k) < 4096$. The performances of both the proposed AFGPC controllers, and the classical GPC controller [Camacho and Bordons, 2007], as well as the respective command signals are presented in Figures 5.3(a) and 5.3(b). In Figure 5.3(a), the value of $R_L(k)$ is presented after multiplying it by a factor of 0.1. For the classical GPC, $N_p = 8$, $N_u = 1$, and $\lambda = 2.5 \times 10^{-3}$ were chosen. These are the best control parameters found experimentally for the classical GPC. The linear model parameters used in the GPC controller were obtained with the Reaction Curve Method from [Camacho and Bordons, 2007]. For the PID controller, the best control parameters were found by the Ziegler-Nichols method [Ziegler and Nichols, 1942]. From the results presented in Figure 5.3, it can be seen that the proposed fuzzy controllers are able to adequately (attain and) control the system output at the desired reference $z_d(k)$. In the sampling intervals $450 < k < 650$ the output is working approximately at full load status. At $k = 650$, when the load disturbance changes from a value of $R_L(k) = 4000$, down to its initial first value of $R_L(k) = 2500$, there is an overshoot at $k = 650$ in the system response. At $k = 650$ the proposed AFGPC (Method 1) comparing with AFGPC (Method 2), shows a better performance by having a smaller overshoot against the unexpected load disturbance changes. But, the AFGPC (Method 2) controller comparing with AFGPC (Method 1) shows a faster response against setpoint changes during all the control process. In the setpoints



(a)



(b)

Figure 5.3: Real-world control of two coupled DC motors: (a) The performance of the proposed adaptive fuzzy GPC (AFGPC) designs, and of the classical GPC, in the presence of load disturbances in the real DC motors process; and (b) the respective applied command signals.

Table 5.2: Comparison of results of the proposed adaptive fuzzy control, AFGPC, methodologies with other methods (PID, DFC, GPC), for the two coupled DC motors test. Note that, here, the $MSE = \frac{1}{T_n}\sum_{t=1}^{T_n}(DO_{ref}(t) - DO(t))^2$ is related to the tracking error rather than to the estimation error that is used in the training of the T-S fuzzy models. T_n is the total number of samples on the test. All values have been multiplied by 10^4.

Methodology	$1/MSE$
PID	3.4994
DFC	4.0138
GPC	3.8896
AFGPC (Method 1)	4.0385
AFGPC (Method 2)	4.2002

tracking results of the GPC, in the interval $200 \leqslant k \leqslant 350$ a steady error can be observed. Numerical $1/MSE$ results (where MSE is the tracking error) of the proposed adaptive fuzzy control methodologies and of the other methods (PID, DFC, GPC) are reported in Table 5.2. As can be seen from Table 5.2, among all controllers, the AFGPC (Method 2) controller shows a better performance in setpoints tracking. The results imply that both AFGPC control methodologies can lead to the adequate control of the process just based on a data set of the process to initialize the T-S fuzzy models. Furthermore, the results show that when comparing with the classical GPC, the proposed adaptive fuzzy controllers with the T-S fuzzy models cause the output $y(k)$ to track the reference signal $z_d(k)$ with more accuracy.

5.4 Conclusion

In this chapter, the methodologies proposed in Chapter 4 to learn a T-S fuzzy model were integrated with a fuzzy GPC controller. Results from both proposed T-S fuzzy identification methodologies, i.e. the identification methodology based on an unsupervised clustering technique, and also the identification methodology based on a hierarchical fuzzy structure, were used for a fuzzy control design. The integration of the proposed adaptive identification methodologies with the GPC results in an effective adaptive predictive fuzzy control methodology. To validate and demonstrate the performance and effectiveness of the proposed algorithms, they were tested in the

problems of controlling the dissolved oxygen in the activated sludge reactor within a WWTP; and on the control of the real-world experimental setup composed of two coupled DC motors. Results have shown that the proposed controller methodologies can control the process using only a data set of the process to initialize the adaptive T-S fuzzy models. The experiments revealed that the proposed AFGPC frameworks outperform typical well-known control methods such as the PID controller, and the classical GPC controller [Camacho and Bordons, 2007]. In most cases/aspects, the AFGPC (Method 2) has shown a better performance when compared with the AFGPC (Method 1). Taking into account the fact that, except in the T-S fuzzy modeling, in the other parts both controllers utilize a similar control structure/procedure in the design and operation, this difference shows that a model of the process plays a very important role in the design of an AFGPC. This in turn confirms the well known fact that, as mentioned throughout this thesis, a model of the process may play an important role in the design of controller; a good control design can result from the use in the controller of a good estimation of the system model.

Chapter 6

Robust Control Design Based on Synergetic Control Theory

Contents

6.1 Introduction / State of the Art

The control of discrete-time linear parameter varying (DT-LPV) plants in the presence of constraints while the plants are affected by unknown disturbances is a very active research area. Until now, control theory for linear parameter varying (LPV) systems has been showing successful application roles in aircrafts [Jiang *et al.*, 2015], energy production systems [Navalkar and van Wingerden, 2015], wafer scanners [Wassink *et al.*, 2005], robotic systems [Vizer *et al.*, 2015], active suspension of vehicles [Pan *et al.*, 2015], and missiles [Feng and Guangren, 2008]. There are good surveys which discuss on the concerns regarding the guarantee of closed-loop stability and constraint satisfaction and also how these concerns can be addressed [Blanchini, 1999; Mayne *et al.*, 2000; Lin and Antsakis, 2009]. Some existing approaches to address the problem of robust control of constrained systems are set invariance approaches [Mayne and Schroeder, 1997; Riverso *et al.*, 2014], reference governors [Gilbert and Kolmanovsky, 2002; Cairano *et al.*, 2015], as well as receding horizon control [Lucia and Tedesco, 2015].

One limitation of existing approaches to control LPV systems is that most of them are able to control uncertain systems only at the origin (see [Pannocchia, 2004] for a discussion). Under this limitation, controllers can be used to achieve steady-state offset-free control, but the setpoints are never allowed to be changed. In practical applications, many times it happens that setpoints need to be changed. Lacking the possibility of changing the setpoints is considered a limitation. Another limitation is present in the important work [Wan and Kothare, 2002], where a robust constrained controller was formulated which was based on the concept of asymptotically stable invariant ellipsoid [Wan and Kothare, 2003]. However, by following this design methodology, the robust controller is limited just to find an elliptical invariant set [Muñoz-Carpintero *et al.*, 2015]. Some works were presented in the literature to overcome these limitations. For example, [Pannocchia, 2004] proposed

a robust control design which uses an auxiliary system to remove offset. The design of the controller was strongly dependent on setpoint changes: in [Pannocchia, 2004], for any setpoint change, new state and input command targets $(\bar{x}_k, \bar{u}_k)$ need to be calculated to serve as a new origin. Additionally, in [Pannocchia, 2004], the presence of disturbances on the plants is not considered.

In [Rotondo *et al.*, 2014] a robust state feedback control method based on a linear matrix inequality (LMI) based approach for uncertain LPV systems was presented. The approach uses the vector of varying parameters to schedule between uncertain linear time invariant (LTI) systems. A robust tube MPC for linear systems with multiplicative uncertainty has been proposed recently in [Fleming *et al.*, 2015]. Before that, the same authors, in [Fleming *et al.*, 2013], have presented a formulation of robust model predictive control which recursively define at each sampling interval intermediate sets that will contain the predicted state trajectory, and from which it is possible to formulate conditions which guarantee satisfaction of the control problem constraints. In addition, a terminal restriction set belonging to a robustly invariant and feasible set takes care of constraints over the infinite horizon beyond a finite prediction horizon. The MPC design in [Fleming *et al.*, 2015] considered multiplicative uncertainty, but was just devoted to track zero setpoints without considering the presense of disturbances. Li *et al.* [2014a] proposed a design of model predictive control (MPC) for LPV systems. A gain scheduling strategy was adopted to handle the system uncertainty and to achieve a fixed learning model for repetitive predictive control (RPC). The design of the RPC in [Li *et al.*, 2014a] has been limited just to periodic step-type disturbances varying between two values. In [Al-Othman and Irving, 2007], a robust state estimator for LPV systems based on maximum constrains satisfaction (MCS) of uncertain measurements and a genetic algorithm was proposed. But the work was limited to state estimation not control design.

One of the main drawbacks of MPC based methodologies is, in general, the computational effort required to solve the constrained finite-time optimal control (CFTOC) problem at each sampling instance. This effort can prevent the application of MPC to systems with a high sampling rate. Some techniques like explicit form of MPC [Besselmann *et al.*, 2012] have been proposed as a remedy for this problem, and as an alternative way to guarantee constraint recursive feasibility and asymptotic stability of closed loop LPV systems. However, due to its complexity,

research on this domain is still a challanging issue [Hajiloo, 2016]. This was a reason to start to investigate on the possibility of combining these two control techniques, MPC and sliding mode control (SMC), to take advantage of the MPC capability of coping with system constraints and the SMC robustness properties [Wang *et al.*, 2013]. However, other common and important drawback of the SMC scheme, or the model predictive SMC design, is the existence of the chattering phenomena caused by the high frequency switching control law which needs to be seriously taken into account [Lee *et al.*, 2009; Feng *et al.*, 2014].

The Russian researcher Kolesnikov [Kolesnikov, 1994] introduced synergetic control theory (SCT) for the first time. To date, synergetic control (SC) theory has been utilized in power systems control [Santi *et al.*, 2003, 2004; Jiang, 2009; Bai *et al.*, 2012]. The SC is an effective technique which can possess the properties of order reduction and decoupling in the design procedure [Santi *et al.*, 2003]. SC is well-suited for digital implementation, and when compared to SMC, it gives better control of the dynamics outside the target manifold, and avoids the chattering phenomenon [Santi *et al.*, 2003, 2004], because the SCT-based controllers are designed without discontinuous part(s). But SCT has never been used for a robust control design. Therefore, the development of a robust control design based on SC theory, and the examination and evaluation of the corresponding robustness properties is an important and promising research subject. In this Chapter, achieving the control of uncertain systems in the presence of disturbance and system constrains, tracking of varying non-zero setpoints, and a chattering-free operation will be a goal.

In this chapter a new robust constrained control methodology for discrete-time linear parameter varying (DT-LPV) systems, is proposed based on a SCT approach. The proposed controller inherits the SMC advantages, while it overcomes the SMC chattering concerns. The proposed SC based design is called the robust discrete SCT controller (RDSCTC). First, a SCT macro-variable is defined based on the output tracking error. To ensure asymptotic convergence of the tracking error to a bounded manifold, a system evolution constraint is given. Through combination of the evolution constraint and macro-variable functions, the dynamics of the error equation is constructed. Solving the dynamic stability equation results in a desired command signal to address the control objectives. It is shown that in systems without uncertainty, and for any unmeasured bounded additive disturbance, the proposed controller accomplishes the goal of stabilizing the system by asymptotically

driving the error of the controlled variable to a bounded set containing the origin and then maintaining it there.

Moreover, given an uncertain DT-LPV system jointly subject to unmeasured and constrained additive disturbances, and constraints in states, input commands, and reference signals (setpoints), then invariant set theory is used to find an appropriate polyhedral robust invariant region in which the proposed control framework is guaranteed to robustly stabilize the closed-loop system. Furthermore, this is achieved even for the case of varying non-zero control setpoints in such uncertain DT-LPV systems. The proposed controller is used to steer the state from any initial state inside the robustly positively invariant set to desired and allowable setpoints. Unlike other control methodologies, such as MPC based control, the proposed controller is characterized to have a simple structure leading to an easy implementation, and a non-complex design process. The effectiveness of the proposed method and the implications of the controller design on feasibility and closed-loop performance are demonstrated through application examples on the temperature control on a Continuous Stirred Tank Reactor (CSTR) plant.

As another case study, a new control design to control nonlinear state dependent discrete time varying (NSDDTV) systems in the presence of unknown additive disturbances, is presented. The proposed discrete-time SCT control framework uses a proportional-integral (PI) type of the SCT macro-variable and a one-step delayed estimation of the disturbance. The proposed controller is characterized to have robustness properties and advantages of SMC, but without inheriting the SMC chattering phenomena. To show the efficiency of the second control design which is based on a PI-type of SCT macro-variable, it is applied on the Escherichia Coli (E. Coli) plant. The cultivation of E. Coli bacteria is widely used by geneticists and biopharmaceuticals to produce medicines and vaccines. In such industries, biotechnological processes are known by their harsh complex environment. A fed-batch bioprocess requires an appropriate controller to suppress the inherent nonlinearity and disturbances. The proposed controller based on a PI-type of SCT macro-variable, is going to maximize the biomass productivity of cultures of E. Coli through acetate concentration control, by manipulating the substrate feed rate, while it is assumed that an additive disturbance affects the bioprocess model.

The Chapter is organized as follows. Section 6.2 presents some introductory notations and definitions. Section 6.3 introduces the proposed online RDSCTC

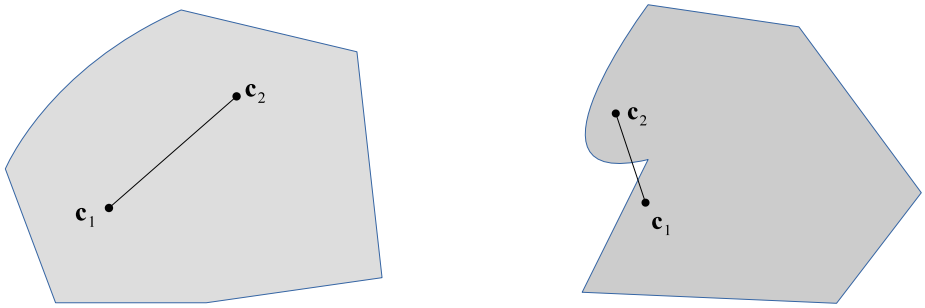


Figure 6.1: Difference between convex and non-convex set: examples of a (a) convex set, and (b) non-convex set.

and presents the RDSCTC stability proof, and then performs robustly positively invariant set constraint analysis for the application of the controller to DT-LPV systems with input and state constraints in the presence of external disturbance. In Section 6.4, an integral type of robust SCT control design for application to NSDDTV systems is presented. In Section 6.5, results of the applications of the RDSCTC controller, and the proposed controller based on a PI-type of SCT macro-variable, their feasibilities and closed-loop performances, are presented and analyzed. Finally, Section 6.6 makes concluding remarks.

6.2 Polytopic Uncertainty, Notations, and Definitions

For the design of the robust constrained control methodology for discrete-time linear parameter varying (DT-LPV) systems that will be proposed in Section 6.3, having relevant knowledge regarding the concepts of polytopic uncertainty in the geometrical space, is necessary. Some useful definitions are given below.

Let $\mathbb{R}$ denote the field of real numbers, and $\mathbb{R}^n$ the n-dimensional Euclidean space. A set C is "convex" if the line segment joining any two points $\mathbf{c}_1$ and $\mathbf{c}_2$ in C remains entirely in C. Given $\mathbf{c}_1$ and $\mathbf{c}_2$ in C and $\lambda \in [0, 1]$, $\lambda\mathbf{c}_1 + (1 - \lambda)\,\mathbf{c}_2$ in C is called a "convex combination" of $\mathbf{c}_1$ and $\mathbf{c}_2$. Figure 6.1 presents examples of convex and nonconvex sets. Given a set C (not necessarily convex) its "convex hull", denoted as $Co\{C\}$, is the smallest convex set which contains C.

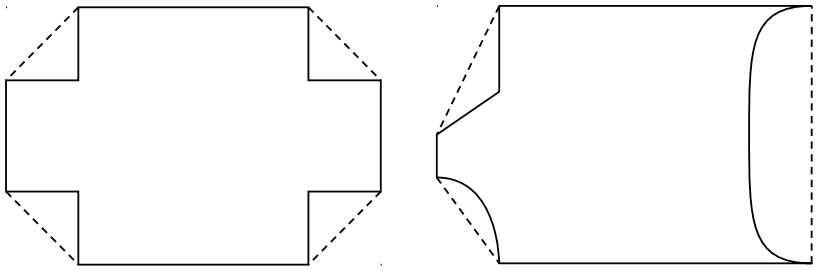


Figure 6.2: Examples of sets and their corresponding convex hulls. The convex hull of a set is the smallest convex set which includes the set.

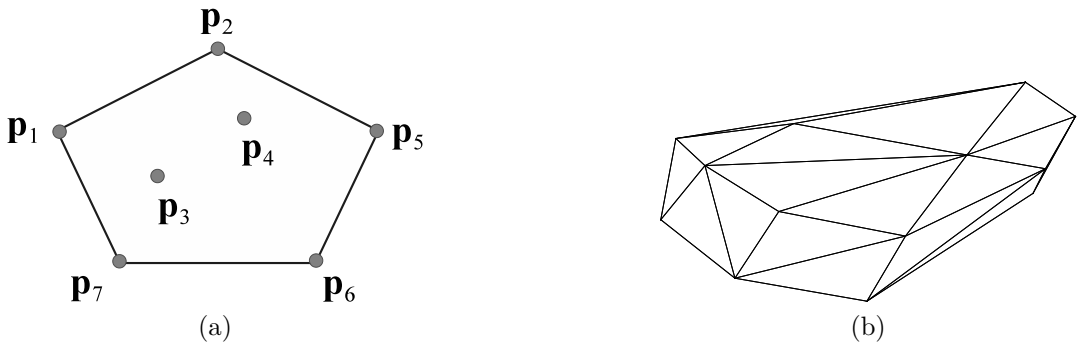


Figure 6.3: Examples of polytopes: (a) an example of 2-D polytope generated by vertices $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_5$, $\mathbf{p}_6$, and $\mathbf{p}_7$, and (b) a 3-D polytope.

Figure 6.2 presents examples of convex hulls. A "polyhedron" is the intersection of a collection of half-spaces. A set is equivalently defined as a "convex polyhedron" if it is the set of solutions to a finite system of linear inequalities. A "polytope" is a bounded polyhedron, or equivalently, the convex hull of a finite set of points $P = \text{convexhull}\{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N\}$. Figure 6.3 presents examples polytopes. In Figure 6.3(a), each $\mathbf{p}_i$, $i = 1, 2, 5, 6, 7$, is a vertex of the polytope $P = \text{convexhull}\{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_5, \mathbf{p}_6, \mathbf{p}_7\}$.

A convex combination of a set of points $\mathbf{p}_i$, $i = 1, \ldots, N$ is a point $\mathbf{p}$ such that:

$$\mathbf{p} = \sum_{i=1}^{N} \lambda_i \mathbf{p}_i, \quad \sum_{i=1}^{N} \lambda_i = 1 \quad \lambda_i \geq 0. \tag{6.1}$$

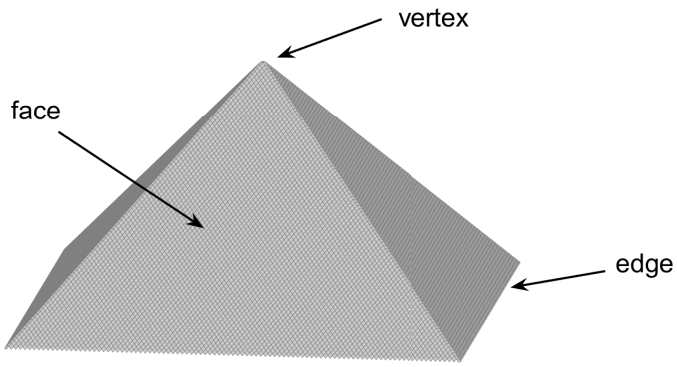


Figure 6.4: 3-D polymap: examples of face, vertex, and edge on a polytope.

The constraint set for $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_N]^T$ defined in (6.1) is called the "unit simplex". A "vertex" of a convex set is a point that cannot be generated as the convex combination of two distinct points of the convex set.

A "face" of a polytope is the intersection of the polytope with a tangent hyperplane. An "edge" (or side) is a 1-D face (a line segment) where two 2-D faces of a polytope meet. A "vertex" (or extreme point) is a 0-D face (a point) at which three or more edges meet (Figure 6.4).

The operation $A \sim B$ represents the Pontryagin difference of two polyhedra $A \subset \mathbb{R}^n$ and $B \subset \mathbb{R}^n$: $A \sim B := \{\mathbf{c} \in \mathbb{R}^n \mid \mathbf{c} + \mathbf{b} \in A, \forall \mathbf{b} \in B\}$. Given a vector $\mathbf{v}$, then $||\mathbf{v}||_\infty$ is the Chebyshev vector norm of $\mathbf{v}$. If the set $\Gamma \subset X \times Y$, then the projection of Γ onto X is defined as $Proj_X(\Gamma) := \{\mathbf{x} \in X \,|\exists \mathbf{y} \in Y \mid (\mathbf{x}, \mathbf{y}) \in \Gamma\}$. The operation $\mathbf{A} * U$ represents the linear map of a polyhedron, where $\mathbf{A}$ is a real matrix and U is a polyhedron. $A \oplus B$ denotes the Minkowski sum of convex polyhedra A and B: $A \oplus B := \{\mathbf{x} \in \mathbb{R}^n \mid \exists\, \mathbf{a} \in A, \mathbf{b} \in B : \mathbf{x} = \mathbf{a} + \mathbf{b}\}$. Given the two sets $\Gamma \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{Q}\mathbf{x} \leqslant \mathbf{q}\}$ and $\Phi \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{S}\mathbf{x} \leqslant \mathbf{s}\}$, $\Gamma \cap \Phi$ is denoted the intersection of convex polyhedra Γ and Φ: $\Gamma \cap \Phi := \left\{\mathbf{x} \in \mathbb{R}^n \;\middle|\; \begin{bmatrix}\mathbf{Q} & \mathbf{S}\end{bmatrix}^T \mathbf{x} \leqslant \begin{bmatrix}\mathbf{q} & \mathbf{s}\end{bmatrix}^T\right\}$.

6.3 Robust Discrete SCT Control Design

In this section a novel robust constrained control methodology for discrete-time linear parameter varying (DT-LPV) systems, is proposed based on a synergetic control theory (SCT) approach. It is shown that in DT-LPV systems without uncertainty, and for any unmeasured bounded additive disturbance, the proposed controller accomplishes the goal of stabilizing the system by asymptotically driving the error of the controlled variable to a bounded set containing the origin and then maintaining it there. Moreover, given an uncertain DT-LPV system jointly subject to unmeasured and constrained additive disturbances, and constraints in states, input commands, and reference signals (setpoints), then invariant set theory is used to find an appropriate polyhedral robust invariant region in which the proposed control framework is guaranteed to robustly stabilize the closed-loop system. Furthermore, this is achieved even for the case of time-varying non-zero control setpoints in such uncertain DT-LPV systems. The controller proposed in this section is termed robust discrete SCT controller (RDSCTC). It is clear that both LPV and LTV systems are time varing systems. However, the system dynamics in LTV systems at each instant time k is known, but in LPV systems it is unknown. This is because, LPV systems contain uncertain parameters in the modelling coefficients.

6.3.1 Problem Formulation and Preliminaries

Consider a DT-LPV system in the normal form with parametric uncertainty, disturbance, and both state and input constraints,

$$\mathbf{x}(k+1) = \mathbf{A}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{w}(k), \tag{6.2}$$

$$\mathbf{y}(k) = \mathbf{C}(k)\mathbf{x}(k), \tag{6.3}$$

$$(\mathbf{A}(k), \mathbf{B}(k)) \in \Gamma \subset \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times m}, \tag{6.4}$$

where $\mathbf{x}(k)$ is the plant state, and $\mathbf{u}(k)$ is the control input, which are subject to

$$\mathbf{x}(k) = [x_1(k), \ldots, x_n(k)]^T \in \mathcal{X} \subset \mathbb{R}^n, \tag{6.5}$$

$$\mathbf{u}(k) = [u_1(k), \ldots, u_m(k)]^T \in \mathcal{U} \subset \mathbb{R}^m. \tag{6.6}$$

$\mathbf{w}(k) = [w_1(k), \ldots, w_n(k)]^T \in \mathcal{W} \subset \mathbb{R}^n$ is a bounded, and unmeasured disturbance, $\mathbf{y}(k) = [y_1(k), \ldots, y_p(k)]^T \in \mathbb{R}^p$ is the output variable, and $\mathbf{C}(k) = [c_{ij}(k)] \in \mathbb{R}^{p \times n}$ is a non-negative matrix (i.e. $c_{ij}(k) \geqslant 0$ for all $i = 1, \ldots, p$, $j = 1, \ldots, n$). $\mathbf{y}(k)$ is also considered the controlled variable, i.e. the variable to be controlled to a given desired setpoint $\mathbf{z}_d(k) \in \mathcal{Z} \subset \mathbb{R}^p$. For polytopic uncertainty, Γ is the polytope $Co\{(\mathbf{A}_1^v, \mathbf{B}_1^v), \ldots, (\mathbf{A}_L^v, \mathbf{B}_L^v)\}$, where $(\mathbf{A}_j^v, \mathbf{B}_j^v)$, $j = 1, \ldots, L$, are the vertices of the convex hull. Any $(\mathbf{A}, \mathbf{B})$ within the convex set Γ is a linear combination, more specifically a convex combination, of the vertices: $\mathbf{A} = \sum_{j=1}^{j=L} \mu(j)\mathbf{A}_j^v$, $\mathbf{B} = \sum_{j=1}^{j=L} \mu(j)\mathbf{B}_j^v$, with $\sum_{j=1}^{j=L} \mu(j) = 1$, and $0 \leqslant \mu(j) \leqslant 1$ (see Section 6.2). Under fairly general conditions a state space model of the form (6.2)-(6.3) can be transformed into an observability, observer, or observable canonical form/realization where $\mathbf{C}(k)$ is non-negative [Sinha and Rózsa, 1976; Antsaklis and Michel, 2007; Tóth, 2010].

Assume that the constraint sets of the state, $\mathcal{X}$, and of the input, $\mathcal{U}$, are in the form of closed and convex polyhedra which are given by some affine inequality constraints on the state and input, i.e. they are convex sets which can be described by a finite number of affine inequality constraints as follows:

$$\mathcal{X} := \{\mathbf{x} \in \mathbb{R}^n \,|\, \mathbf{A}_x\mathbf{x} \leqslant \mathbf{b}_x\}, \tag{6.7}$$

$$\mathcal{U} := \{\mathbf{u} \in \mathbb{R}^m \,|\, \mathbf{A}_u\mathbf{u} \leqslant \mathbf{b}_u\}, \tag{6.8}$$

where $\mathbf{A}_x \in \mathbb{R}^{q_x \times n}$, $\mathbf{A}_u \in \mathbb{R}^{q_u \times m}$, $\mathbf{b}_x \in \mathbb{R}^{q_x}$, and $\mathbf{b}_u \in \mathbb{R}^{q_u}$. It is assumed that $\mathcal{X}$, and $\mathcal{U}$ contain the origin in their interiors. It is assumed that, at each time instant k, the disturbance is unknown, but that the disturbance sequence $\mathbf{w}(\cdot)$ takes on values $\mathbf{w}(k) \in \mathcal{W} \subset \mathbb{R}^n$ for all $k \in \mathbb{N}$.

Similarly to (6.7) and (6.8), $\mathcal{W}$ and $\mathcal{Z}$ are assumed to be polyhedral sets with non-empty interiors containing the origin, given by a finite number of affine inequality constraints as follows:

$$\mathcal{W} := \{\mathbf{w} \in \mathbb{R}^n \,|\, \mathbf{A}_w\mathbf{w} \leqslant \mathbf{b}_w\}, \tag{6.9}$$

$$\mathcal{Z} := \{\mathbf{z}_d \in \mathbb{R}^p \,|\, \mathbf{A}_z\mathbf{z}_d \leqslant \mathbf{b}_z\}, \tag{6.10}$$

where $\mathbf{A}_w \in \mathbb{R}^{q_w \times n}$, $\mathbf{A}_z \in \mathbb{R}^{q_z \times p}$, $\mathbf{b}_w \in \mathbb{R}^{q_w}$, and $\mathbf{b}_z \in \mathbb{R}^{q_z}$.

Some fundamental statements which will be important in the sequel, are listed below.

Definition 6.3.1. A set $\mathcal{O}_\infty \subset \mathbb{R}^n$ is a *robust control invariant set* for the system (6.2)-(6.4) if $\forall \mathbf{x}(k) \in \mathcal{O}_\infty$, there exists an auxiliary control signal $\mathbf{u}(k) = \mathbf{h}(\mathbf{x}(k))$, such that $\forall \mathbf{w}(k) \in \mathcal{W}$, $\forall \mathbf{z}_d(k) \in \mathcal{Z}$, $\forall (\mathbf{A}, \mathbf{B}) \in \Gamma$, $\forall k \in \mathbb{N}^+$, the following holds: $\mathbf{x}(k+1) \in \mathcal{O}_\infty$, $\forall k \in \mathbb{N}^+$.

Assumption 6.3.1. *There exists a robust control invariant set $\mathcal{O}_\infty$ for the system (6.2)-(6.3) with the control law to calculate the command signal $\mathbf{u}(k) \in \mathcal{U}$, such that $\mathcal{O}_\infty$ lies in the interior of Ξ, where Ξ is the set of all the states for which the constraints on the plant states, and on the corresponding plant inputs $\mathbf{u} = \mathbf{u}(\mathbf{x}, \mathbf{z}_d, \mathbf{w})$, are satisfied for any choice of the desired outputs $\mathbf{z}_d \in \mathcal{Z}$.*

Assumption 6.3.2. [*General*] *The measurement of the plant state is available at each sample time k, system $\{\mathbf{A}(k), \mathbf{B}(k), \mathbf{C}(k)\}$ is output controllable, and $\mathbf{C}(k) = [c_{ij}(k)] \in \mathbb{R}^{p \times n}$ is a non-negative matrix (i.e. $c_{ij}(k) \geqslant 0$ for all k, and all $i = 1, \ldots, p$, $j = 1, \ldots, n$).*

Definition 6.3.2. A system is asymptotically ultimately bounded if, after starting at a set of bounded initial conditions, the system converges asymptotically to a bounded set.

Next, it is presented a useful result [Kerrigan and Maciejowski, 2004; Bemporad *et al.*, 2003; Pannocchia and Kerrigan, 2005] which will also be helpful for the present work:

Proposition 6.3.1. *For the polyhedron $\mathcal{J}$ which is given by*

$$\mathcal{J} := \{\mathbf{g} \in \mathbb{R}^t \,|\, \mathbf{F}\mathbf{g} \leqslant \mathbf{v} + \mathbf{E}\mathbf{w}, \forall \mathbf{w} \in \mathcal{W}\},$$

where $\mathbf{F} \in \mathbb{R}^{q \times t}$, $\mathbf{E} \in \mathbb{R}^{q \times s}$, $\mathbf{v} \in \mathbb{R}^q$, q, t, and s are positive integers, and $\mathcal{W}$ is a compact (i.e. closed and bounded) subset of $\mathbb{R}^s$, then the following holds

$$\mathcal{J} = \{\mathbf{g} \in \mathbb{R}^t \,|\, \mathbf{F}\mathbf{g} \leqslant \mathbf{v} + \min_{\mathbf{w} \in \mathcal{W}} \mathbf{E}\mathbf{w}\},$$

where the minimization is performed row-wise, i.e. if $\mathbf{e}_i$ denotes the i-th row of $\mathbf{E}$, then $\min_{\mathbf{w} \in \mathcal{W}} \mathbf{E}\mathbf{w} = \left[\min_{\mathbf{w} \in \mathcal{W}} \mathbf{e}_1\mathbf{w}, \ldots, \min_{\mathbf{w} \in \mathcal{W}} \mathbf{e}_q\mathbf{w}\right]^T$. Furthermore, if

$$\mathcal{W} := \{\mathbf{w} \in \mathbb{R}^s \,|\, \| \mathbf{w} \|_\infty \leqslant \gamma\},$$

then

$$\mathcal{J} := \{\mathbf{g} \in \mathbb{R}^t \,|\, \mathbf{F}\mathbf{g} \leqslant \mathbf{v} - \gamma \mathbf{E}_s\},$$

where $\mathbf{E}_s = \text{abs}(\mathbf{E})\,[1_1\ 1_2 \ldots 1_s]^T$.

$\text{abs}(\mathbf{E}) = [\ |e_{ij}|\ ]$ represents the matrix of the absolute values of the corresponding elements of matrix $\mathbf{E} = [e_{ij}]$, and $1_i = 1$, for $i = 1, \ldots, s$.

Remark 1. *The* $\mathbf{A}(k)$*,* $\mathbf{B}(k)$*, and* $\mathbf{C}(k+1)$ *matrices can be time-varying, but to simplify the notation, in the sequel they will be simply denoted by* $\mathbf{A}$*,* $\mathbf{B}$*, and* $\mathbf{C}$*, respectively.*

Under the above assumptions and definitions, this section presents a novel robust controller for DT-LPV systems. It is shown that in the plant without uncertainty and for any bounded disturbance, the controller accomplishes the goal of asymptotically driving the error of the controlled variable to a bounded set and then maintaining it there, while driving the controlled variable to any given allowable setpoints. Then, considering the case of the presence of plant uncertainty, as well as known constraints on the state variables, input commands, reference signals, and disturbances, it is derived a robustly positively invariant set for the proposed controller.

6.3.2 Synergetic Control Theory Principles

Synergetic control theory (SCT) offers a control framework based on a theory conceived to control non-linear dynamical processes [Kolesnikov, 2014], [Kolesnikov, 1994, cited in [Kolesnikov, 2014]]. The SCT based controllers are coordinated with the internal characteristics of the systems. The synergetic control structure is designed to make the closed-loop control system converge to regions/sets of attraction that correspond to the control purposes, and then force the trajectories to stay on those regions.

Depending on the dimensionality of the state space, attractors can be points, contours, tori or regions of fractal dimensionality. Generating such regions of attraction, or attractors, in dynamical systems, is one of the main concepts of the SCT. The SCT is attractive because, unlike approaches such as SMC, the derived controllers are continuous, and because controllers can be derived from a first-order

differential equation [Liu and Hsiao, 2012], [Nusawardhana *et al.*, 2007]. Convergence can be analyzed by using stability methods such as the Lyapunov Stability Theory [Kalman and Bertram, 1960; Sastry, 1999; Liu and Hsiao, 2012; Guang-Yue *et al.*, 2013], or using a potential function of the system [Olfati-Saber and Murray, 2002].

The control purposes in SCT are formulated as a collection of aggregated macro-variables ψ_i that need to be zero. The aggregated macro-variables, ψ_i, $i = 1, \ldots, r \leqslant m$, are functions of system variables $\mathbf{x}$, $\mathbf{z}_d$, and control signals $\mathbf{u}$: $\psi_i = \psi_i(\mathbf{x}, \mathbf{z}_d, \mathbf{u}) = \psi_i(k) = \psi_i(\mathbf{x}(k), \mathbf{z}_d(k), \mathbf{u}(k))$. Hence, to attain the control purposes, attractors are introduced where all the aggregated macro-variables are equal to zero. The SCT gives an equation which can be used for creating dynamical systems with attractors at $\psi_i = 0$ which is expressed as

$$T\dot{\boldsymbol{\psi}} + \boldsymbol{\phi}(\boldsymbol{\psi}) = 0, \tag{6.11}$$

where T determines the rate of convergence to the attractor, $\boldsymbol{\psi} = [\psi_1, \psi_2, \ldots, \psi_r]^T$ is the vector of aggregated macro-variables, $\dot{\boldsymbol{\psi}}$ is the derivative of the aggregated macro-variable (vector) with respect to time, and $\boldsymbol{\phi}(\cdot)$ is some function that makes the solution of (6.11) approach the attractor.

One of the possible expressions for $\boldsymbol{\phi}(\cdot)$ is $\boldsymbol{\phi}(\boldsymbol{\psi}) = \boldsymbol{\psi}$. In this case, equation (6.11) is rewritten as:

$$T\dot{\boldsymbol{\psi}} + \boldsymbol{\psi} = 0. \tag{6.12}$$

Assumption 6.3.3. *In this chapter, it is assumed that the process to be controlled is a single-input single-output (SISO) plant, and* $r = m = p = 1$.

6.3.3 Proposed Controller

In this section, the discrete time controller based on synergetic control theory is designed in the presence of unmeasured bounded additive disturbances without constraints on the state $\mathbf{x}$, and input $\mathbf{u}$ of the DT-LPV system [Rastegar *et al.*, 2017c]. Subsrection 6.3.5, will present the analysis of the proposed controller for systems which are also subject to constraints in states, input commands, reference signals, and unmeasured constrained additive disturbances.

The SCT can be useful for a robust control design [Rastegar *et al.*, 2016a, 2017d]

in such a way that the constructed controller can address the main control purposes investigated in this section. In fact, the SCT gives an attractor which was defined in (6.12). By using this attractor, the dynamic error (6.17) of the system will be formed. Then, with a simple analytic procedure, the control law (6.18)-(6.19) will be derived, and can then be easily implemented.

For the first phase of the work, let the tracking error be defined as $\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{z}_d(k)$, where $\mathbf{z}_d(k)$ contains the setpoints, and define the macro-variable $\boldsymbol{\psi}(k) = \mathbf{e}(k)$:

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{z}_d(k), \tag{6.13}$$

$$\boldsymbol{\psi}(k) = \mathbf{e}(k), \tag{6.14}$$

$$\boldsymbol{\psi}(k+1) = \mathbf{e}(k+1). \tag{6.15}$$

Replacing (6.14) and (6.15) into the discrete version of (6.12), discretized using the forward differences method, yields

$$T\frac{\mathbf{e}(k+1) - \mathbf{e}(k)}{T_s} + \mathbf{e}(k) = 0, \tag{6.16}$$

where T_s is the sampling time period. Eq. (6.16) describes the dynamic error of the system as

$$T\mathbf{e}(k+1) - (T - T_s)\mathbf{e}(k) = 0. \tag{6.17}$$

Using (6.2), and (6.13), the error dynamics (6.17) can be rewritten as:

$$T[\mathbf{CAx}(k) + \mathbf{CBu}(k) + \mathbf{Cw}(k) - \mathbf{z}_d(k+1)] - (T - T_s)\mathbf{e}(k) = 0. \tag{6.18}$$

In this Section, it is used the notation that $\bar{\boldsymbol{\varrho}}$ is the nominal value, or the estimated value, of the real value $\boldsymbol{\varrho}$, for any $\boldsymbol{\varrho}$, i.e. $\bar{\boldsymbol{\varrho}}$ is the value of $\boldsymbol{\varrho}$ that is used in the controller.

Assumption 6.3.4. *It is assumed that* $\det(\bar{\mathbf{C}}\bar{\mathbf{B}}) \neq 0$.

Some points are worth to be remarked regarding the mathematical implications of the assumption that $\det(\mathbf{CB}) \neq 0$. First, there are many relevant practical situations in which the condition $\det(\mathbf{CB}) \neq 0$ can be satisfied on the plants. Second, note that $\det(\mathbf{CB}) \neq 0$ implies that $\mathbf{CB}$ should be a square matrix in order to be

possible to calculate its determinant, which in turn implies that $m = p$. In this chapter, the process to be controlled is single-input single-output (SISO), $m = p = 1$ (Assumption 6.3.3).

From (6.3), (6.13), (6.14), and (6.18), and Assumption 6.3.4, the input signal is obtained as

$$\begin{aligned}\mathbf{u}(k) &= \left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1}\left[-T\bar{\mathbf{C}}\bar{\mathbf{A}}\mathbf{x}(k) - T\bar{\mathbf{C}}\mathbf{w}(k) + T\mathbf{z}_d(k+1) + (T-T_s)\boldsymbol{\psi}(k)\right], \\ &= \left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1}\left[-T\bar{\mathbf{C}}\bar{\mathbf{A}}\mathbf{x}(k) - T\bar{\mathbf{C}}\mathbf{w}(k)\right. \\ &\qquad \left. + T\mathbf{z}_d(k+1) + (T-T_s)(\bar{\mathbf{C}}\mathbf{x}(k) - \mathbf{z}_d(k))\right]. \quad (6.19)\end{aligned}$$

The command law in Eq. (6.19) needs to have an explicit form. By this reason, and assuming for the sake of motivation that $\mathbf{w}(k)$ is smooth so that the change of $\mathbf{w}(k)$ in one sampling interval is small, $\mathbf{w}(k) \approx \mathbf{w}(k-1)$, then the unknown bounded disturbance $\mathbf{w}(k) \in \mathcal{W}$ is replaced by $\bar{\mathbf{w}}(k-1)$ which is the one-step delayed estimation of $\mathbf{w}(k)$ [Xu and Li, 2012] obtained from (6.2), and projected into the closest point in $\mathcal{W}$. Specifically,

$$\bar{\mathbf{w}}(k-1) = \arg\min_{\mathbf{w}' \in \mathcal{W}} ||\mathbf{w}' - \hat{\mathbf{w}}(k-1)||, \quad (6.20)$$

$$\hat{\mathbf{w}}(k-1) = \mathbf{x}(k) - \bar{\mathbf{A}}(k-1)\mathbf{x}(k-1) - \bar{\mathbf{B}}(k-1)\mathbf{u}(k-1), \quad (6.21)$$

where $||\cdot||$ is the Euclidean distance. Note that $\bar{\mathbf{w}}(k-1) \neq \hat{\mathbf{w}}(k-1)$ only if $\hat{\mathbf{w}}(k-1) \notin \mathcal{W}$, which in turn may only happen if $\bar{\mathbf{A}} \neq \mathbf{A}$ or $\bar{\mathbf{B}} \neq \mathbf{B}$. Also note that, if $\bar{\mathbf{A}} = \mathbf{A}$ and $\bar{\mathbf{B}} = \mathbf{B}$, then, from (6.2), $\hat{\mathbf{w}}(k-1) = \mathbf{w}(k-1) \in \mathcal{W}$, and from (6.20)-(6.21), $\bar{\mathbf{w}}(k-1) = \hat{\mathbf{w}}(k-1)$:

$$\bar{\mathbf{w}}(k-1) = \mathbf{x}(k) - \bar{\mathbf{A}}(k-1)\mathbf{x}(k-1) - \bar{\mathbf{B}}(k-1)\mathbf{u}(k-1). \quad (6.22)$$

After obtaining $\bar{\mathbf{w}}(k-1) \in \mathcal{W}$ from (6.20), then, Eq. (6.19) is transformed into:

$$\mathbf{u}(k) = \bar{\mathbf{A}}_1\mathbf{x}(k) + \bar{\mathbf{A}}_2\mathbf{z}_d(k+1) + \bar{\mathbf{A}}_3\mathbf{z}_d(k) + \bar{\mathbf{A}}_4\bar{\mathbf{w}}(k-1), \quad (6.23)$$

where $\bar{\mathbf{A}}_1$ and $\bar{\mathbf{A}}_4$ are $1 \times n$ matrices and $\bar{\mathbf{A}}_2$ and $\bar{\mathbf{A}}_3$ are scalar values defined as:

$$\begin{aligned}
\bar{\mathbf{A}}_1 &= -\left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1}\left\{T\bar{\mathbf{C}}\bar{\mathbf{A}} + (T_s - T)\bar{\mathbf{C}}\right\}, \\
\bar{\mathbf{A}}_2 &= \left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1} T, \\
\bar{\mathbf{A}}_3 &= \left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1} (T_s - T), \\
\bar{\mathbf{A}}_4 &= \left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1} (-T\bar{\mathbf{C}}).
\end{aligned} \tag{6.24}$$

In situations where it is not possible to predict $\mathbf{z}_d(k+1)$, then the command signal $\mathbf{u}(k)$ in (6.19), (6.23), can be approximated/estimated, by using $\mathbf{z}_d(k+1) \approx 2\mathbf{z}_d(k) - \mathbf{z}_d(k-1)$ to extrapolate for the next future reference signal, $\mathbf{z}_d(k+1)$.

The stability of the closed-loop system (6.2)-(6.3), (6.19)-(6.24), can be attained as will be discussed below in Subsections 6.3.4, and 6.3.5 [Rastegar *et al.*, 2017c]. In Subsection 6.3.4 it will be shown that, by adequately choosing the controller parameters T, and T_s, the control signal (6.19)-(6.24) enforces the state variables of the closed-loop system without uncertainty in $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, and with unmeasured bounded disturbances, to asymptotically converge to a macro-variable manifold, after starting from any initial state, and then to cause the state variables to remain in a bounded region near the macro-variable manifold. Then, in Subsection 6.3.5, a robustly positively invariant set for the uncertain closed loop system subject to constraints in the system variables, will be derived.

6.3.4 Robust Closed-Loop Stability

In the stability analysis performed in this subsection, it will be assumed that the nominal plant parameter values used in the controller are equal to the real plant parameters, i.e. $\bar{\boldsymbol{\varrho}} = \boldsymbol{\varrho}, \forall \boldsymbol{\varrho} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$. From (6.20)-(6.21), (6.3), and (6.24), this implies that $\bar{\boldsymbol{\varrho}} = \boldsymbol{\varrho}, \forall \boldsymbol{\varrho} \in \{\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3, \mathbf{A}_4, \mathbf{w}\}$. Also, in this subsection, the constraints in the system states, input commands, and reference signals, are not considered. In Subsection 6.3.5 these assumptions will be abandoned.

In the design of the SCT control law, the SCT convergence law in Eq. (6.12) must guarantee that after starting from any initial state, the error trajectory monotonically moves towards a SCT macro-variable manifold, also called the robust macro-variable manifold band (*RMMB*). After the error trajectory has reached the *RMMB*,

then in each subsequent time step, the trajectory is confined to remain inside the *RMMB*. It will be shown that the following *RMMB* meets these conditions:

$$RMMB = \left\{ \mathbf{x} \in \mathbb{R}^n : \ |\boldsymbol{\psi}(\mathbf{x}, \mathbf{z}_d)| \leqslant M_\psi = \bar{\mathbf{C}}(\mathbf{w}_{max} - \mathbf{w}_{min}) \frac{T}{2T - T_s} \right\}. \tag{6.25}$$

T and T_s are positive constant values, $\bar{\mathbf{C}}$ contains non-negative values, and $\mathbf{w}_{max} \in \mathbb{R}^n$ and $\mathbf{w}_{min} \in \mathbb{R}^n$ are the row-wise maximum and minimum of $\mathbf{w}(k)$, respectively.

The control law that enforces the SCT convergence law in Eq. (6.12) is given by (6.19), which is then developed as specified in (6.20)-(6.24). Therefore, from (6.20)-(6.24), the control law can be rewritten as follows:

$$\begin{aligned} \mathbf{u}(k) &= \left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1} \left\{ -T\bar{\mathbf{C}}\bar{\mathbf{A}}\mathbf{x}(k) - T\bar{\mathbf{C}}\bar{\mathbf{w}}(k-1) + T\mathbf{z}_d(k+1) + (T - T_s)\boldsymbol{\psi}(k) \right\}, \\ &= \left(T\bar{\mathbf{C}}\bar{\mathbf{B}}\right)^{-1} \left\{ \left[-T\bar{\mathbf{C}}\bar{\mathbf{A}} + (T - T_s)\bar{\mathbf{C}} \right] \mathbf{x}(k) + T\mathbf{z}_d(k+1) \right. \\ &\qquad \left. + (T_s - T)\mathbf{z}_d(k) - T\bar{\mathbf{C}}\bar{\mathbf{w}}(k-1) \right\}. \end{aligned} \tag{6.26}$$

where $T > T_s/2$, and $\det(\bar{\mathbf{C}}\bar{\mathbf{B}}) \neq 0$.

Theorem 6.3.1. *Consider the system (6.2)-(6.3) with control law (6.26). Suppose the reference/desired output variables are given by $\mathbf{z}_d(k)$, and the dynamic error is defined as $\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{z}_d(k)$. Also assume that $T < T_s < 2T$, $\det\left(\bar{\mathbf{C}}\bar{\mathbf{B}}\right) \neq 0$, and $\bar{\boldsymbol{\varrho}} = \boldsymbol{\varrho}, \forall \boldsymbol{\varrho} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$. Then, the closed-loop system (6.2)-(6.3) in the presence of unknown bounded disturbance $\mathbf{w}(k)$ is stable, and by using the control law (6.26) the error trajectory asymptotically converges to the RMMB (6.25). After the trajectory has reached the RMMB, then in each subsequent time step, the trajectory is confined to remain inside the RMMB.*

Proof. First, note that several steps of the proof are justified by the fact that $\bar{\boldsymbol{\varrho}} = \boldsymbol{\varrho}, \forall \boldsymbol{\varrho} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$. Since $\bar{\boldsymbol{\varrho}} = \boldsymbol{\varrho}, \forall \boldsymbol{\varrho} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$, then from (6.20)-(6.21), and (6.2),

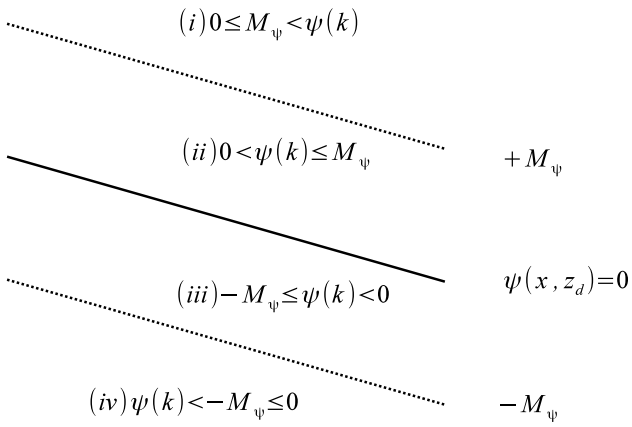


Figure 6.5: Illustration of the *RMMB* function and four possible conditions for $\boldsymbol{\psi}(k)$.

$\bar{\mathbf{w}}(k-1) = \mathbf{w}(k-1)$ is given by (6.22). Then, using (6.26), note that

$$\begin{aligned}
&\boldsymbol{\psi}(k+1) = \mathbf{e}(k+1) \\
&= \mathbf{y}(k+1) - \mathbf{z}_d(k+1) \\
&= \mathbf{CAx}(k) + \mathbf{CBu}(k) + \mathbf{Cw}(k) - \mathbf{z}_d(k+1), \\
&= \mathbf{CAx}(k) + \mathbf{CB}(T\bar{\mathbf{C}}\bar{\mathbf{B}})^{-1}\left[-T\bar{\mathbf{C}}\bar{\mathbf{A}}\mathbf{x}(k) - T\bar{\mathbf{C}}\bar{\mathbf{w}}(k-1)\right. \\
&\quad \left. + T\mathbf{z}_d(k+1) - (T_s - T)\boldsymbol{\psi}(k)\right] + \mathbf{Cw}(k) - \mathbf{z}_d(k+1), \\
&= \mathbf{CAx}(k) - \bar{\mathbf{C}}\bar{\mathbf{A}}\mathbf{x}(k) - \bar{\mathbf{C}}\bar{\mathbf{w}}(k-1) + \mathbf{z}_d(k+1) + \left(1 - \frac{T_s}{T}\right)\boldsymbol{\psi}(k) + \mathbf{Cw}(k) \\
&\quad - \mathbf{z}_d(k+1) = \frac{T - T_s}{T}\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)).
\end{aligned} \tag{6.27}$$

The proof will be made as follows: (a) if $\boldsymbol{\psi}(k) \notin RMMB$, then the discrete-time Lyapunov function [Haddad and Chellaboina, 2008] $V(k) = \boldsymbol{\psi}^2(k) := \boldsymbol{\psi}^T(k)\boldsymbol{\psi}(k)$ is chosen, and it is shown that $\Delta V = V(k+1) - V(k) < 0$, i.e. $\boldsymbol{\psi}^2(k+1) < \boldsymbol{\psi}^2(k)$; and (b) it is shown that if $\boldsymbol{\psi}(k) \in RMMB$, then $\boldsymbol{\psi}(k+1) \in RMMB$.

Based on the definition of the *RMMB* in Eq. (6.25), $\boldsymbol{\psi}(k)$ can meet one of the following four different conditions: (i) $0 \leqslant M_\psi < \boldsymbol{\psi}(k)$, (ii) $0 < \boldsymbol{\psi}(k) \leqslant M_\psi$, (iii) $-M_\psi \leqslant \boldsymbol{\psi}(k) < 0$, (iv) $\boldsymbol{\psi}(k) < -M_\psi \leqslant 0$. For each of these four conditions, both (a) and (b) must be satisfied. Figure 6.5 presents all possible conditions. These four conditions are analyzed in detail below.

(i). If $\mathbf{0} \leqslant \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min})\frac{T}{2T-T_s} < \boldsymbol{\psi}(k)$, then from (6.27)

$$\begin{aligned}
\boldsymbol{\psi}(k+1) - \boldsymbol{\psi}(k) &= \left(\frac{T-T_s}{T} - 1\right)\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \\
&= -\frac{T_s}{T}\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \\
&\leqslant -\frac{T_s}{T}\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) \\
&< -\frac{T_s}{T}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min})\frac{T}{2T-T_s} + \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) \\
&= 2\frac{T-T_s}{2T-T_s}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) < \mathbf{0}, \qquad (6.28) \\
\boldsymbol{\psi}(k+1) + \boldsymbol{\psi}(k) &= \left(\frac{T-T_s}{T} + 1\right)\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \\
&= \left(\frac{2T-T_s}{T}\right)\boldsymbol{\psi}(k) + \mathbf{C}\left(\mathbf{w}(k) - \mathbf{w}(k-1)\right) \\
&> \left(\frac{2T-T_s}{T}\right)\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min})\frac{T}{2T-T_s} \\
&\quad - \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) = \mathbf{0}. \qquad (6.29)
\end{aligned}$$

Therefore, from (6.28) and (6.29), $\boldsymbol{\psi}^2(k+1) < \boldsymbol{\psi}^2(k)$.

(ii). If $\mathbf{0} < \boldsymbol{\psi}(k) \leqslant \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min})\frac{T}{2T-T_s}$, then the condition which must be satisfied is $|\boldsymbol{\psi}(k+1)| \leqslant \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min})\frac{T}{2T-T_s}$. Considering also (6.27), and the fact that $T - T_s < 0$, it is noted that

$$\begin{aligned}
\frac{T-T_s}{2T-T_s}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) &\leqslant \boldsymbol{\psi}(k+1) \\
< \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)). \qquad (6.30)
\end{aligned}$$

From (6.30), and considering that $|\mathbf{w}(k) - \mathbf{w}(k-1)| \leqslant \mathbf{w}_{max} - \mathbf{w}_{min}$, and $2T - T_s < T$, then

$$\begin{aligned}
\boldsymbol{\psi}(k+1) &< \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \leqslant \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) \\
&< \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min})\frac{T}{2T-T_s}, \qquad (6.31) \\
\boldsymbol{\psi}(k+1) &\geqslant \frac{T-T_s}{2T-T_s}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1))
\end{aligned}$$

$$\geqslant \frac{T-T_s}{2T-T_s}\mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min}) - \mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min})$$
$$= -\frac{T}{2T-T_s}\mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min}). \tag{6.32}$$

From (6.31) and (6.32),

$$|\boldsymbol{\psi}(k+1)| \leqslant \mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min})\frac{T}{2T-T_s}. \tag{6.33}$$

From (6.33), it is seen that after the trajectory has reached the upper positive side band of the *RMMB*, then in each subsequent time step, the trajectory is confined to remain inside the *RMMB*. Also note that in the case that $\boldsymbol{\psi}(k)=\mathbf{0}$, then (6.31), (6.32), and (6.33), still hold, except that the left most inequality in (6.31) becomes an equality. Therefore, the theorem also holds for $\boldsymbol{\psi}(k)=\mathbf{0}$.

(*iii*). If $-\mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min})\frac{T}{2T-T_s} \leqslant \boldsymbol{\psi}(k) < \mathbf{0}$, then the condition that must be satisfied is (6.33). Considering also that $T < T_s < 2T$, then $\boldsymbol{\psi}(k) < \frac{T-T_s}{T}\boldsymbol{\psi}(k)$. Therefore, from (6.27),

$$\begin{aligned}
\boldsymbol{\psi}(k+1) &= \frac{T-T_s}{T}\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k)-\mathbf{w}(k-1)) \\
&> \mathbf{C}(\mathbf{w}(k)-\mathbf{w}(k-1)) \geqslant -\mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min}) \\
&> -\frac{T}{2T-T_s}\mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min}), \qquad (6.34)\\
\boldsymbol{\psi}(k+1) &= \frac{T-T_s}{T}\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k)-\mathbf{w}(k-1)) \\
&\leqslant -\frac{T-T_s}{T}\mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min})\frac{T}{2T-T_s} + \mathbf{C}(\mathbf{w}(k)-\mathbf{w}(k-1)) \\
&\leqslant -\frac{T-T_s}{2T-T_s}\mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min}) + \mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min}) \\
&= \mathbf{C}(\mathbf{w}_{max}-\mathbf{w}_{min})\frac{T}{2T-T_s}. \qquad (6.35)
\end{aligned}$$

Thus, from (6.34) and (6.35), it is seen that (6.33) holds.

(iv). If $\boldsymbol{\psi}(k) < -\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min})\frac{T}{2T-T_s} \leqslant \mathbf{0}$, then from (6.27),

$$\begin{aligned}
\boldsymbol{\psi}(k+1) - \boldsymbol{\psi}(k) &= \frac{-T_s}{T}\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \\
&> \frac{T_s}{2T-T_s}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \\
&\geqslant \frac{T_s}{2T-T_s}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) - \mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) \\
&= \frac{2(T_s-T)}{2T-T_s}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) > \mathbf{0}, \qquad (6.36) \\
\boldsymbol{\psi}(k+1) + \boldsymbol{\psi}(k) &= \frac{2T-T_s}{T}\boldsymbol{\psi}(k) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \\
&< -\frac{2T-T_s}{T} \times \frac{T}{2T-T_s}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) \\
&\quad + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \\
&= -\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) + \mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) < \mathbf{0}. \qquad (6.37)
\end{aligned}$$

From (6.36) and (6.37) it is concluded that $\boldsymbol{\psi}^2(k+1) < \boldsymbol{\psi}^2(k)$. □

Note that Theorem 6.3.1, may still be useful in the study of the stability of systems with additive uncertainty in $\mathbf{A}(k)$ and $\mathbf{B}(k)$, if it is possible that such uncertainty is first transferred to $\mathbf{w}(k)$ and the system remains in the hypothesis of the theorem regarding $\mathbf{w}(k)$.

6.3.5 Robustly Positively Invariant Set

In this section, the controller presented in Subsection 6.3.3 is applied to DT-LPV systems exposed to constraints in the states, in the input command signals, in the reference signals, and in the unmeasured additive disturbances. Additionally, it is abandoned the assumption that $\bar{\boldsymbol{\varrho}} = \boldsymbol{\varrho}, \forall \boldsymbol{\varrho} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$ which has been made in Subsection 6.3.4, and the problem of computing the robustly positively invariant set of the system is considered.

Assume that the controller is such that it generates the plant inputs $\mathbf{u}(k) = \mathbf{u}(\mathbf{x}(k), \mathbf{z}_d(k), \mathbf{z}_d(k+1), \bar{\mathbf{w}}(k))$, as given by (6.23). Then, let the constraint-admissible set Ξ be defined as the set of all the states $\mathbf{x}(k)$, for which the constraints on the plant states, and on the corresponding plant inputs $\mathbf{u}(k) = \mathbf{u}(\mathbf{x}(k), \mathbf{z}_d(k), \mathbf{z}_d(k+1), \bar{\mathbf{w}}(k))$,

are satisfied for any choice of the desired outputs $\mathbf{z}_d(k) \in \mathcal{Z}$:

$$\Xi = \{\mathbf{x}(k) \in \mathbb{R}^n \,|\, \mathbf{x}(k) \in \mathcal{X} \text{ and } \mathbf{u}(\mathbf{x}(k), \mathbf{z}_d(k), \mathbf{z}_d(k+1)) \in \mathcal{U}, \\ \forall \mathbf{z}_d(k) \in \mathcal{Z}, \forall \mathbf{z}_d(k+1) \in \mathcal{Z}\}. \tag{6.38}$$

Since $\mathcal{X}$ and $\mathcal{U}$ are polyhedra given by affine inequalities, then Ξ is computed by applying Proposition 6.3.1 to the above definition (6.38).

Let $\mathbf{z}_d(\cdot)$ and $\mathbf{w}(\cdot)$ denote the output reference sequence and the disturbance sequence, respectively, where a sequence is defined as an ordered set. Let, $\boldsymbol{\varphi}(\mathbf{x}(0), \mathbf{z}_d(\cdot), \mathbf{w}(\cdot), \cdot)$ be the solution sequence to (6.2), under controller (6.20)-(6.21), (6.23), and reference and disturbance sequences $\mathbf{z}_d(\cdot)$ and $\mathbf{w}(\cdot)$, after starting at initial state $\mathbf{x}(0)$. Also let $\boldsymbol{\varphi}(\mathbf{x}(0), \mathbf{z}_d(\cdot), \mathbf{w}(\cdot), k) = \mathbf{x}(k)$ be the value of $\boldsymbol{\varphi}(\mathbf{x}, \mathbf{z}_d(\cdot), \mathbf{w}(\cdot), \cdot)$ at time k. The constraint admissible robustly positively invariant set $\mathcal{O}_\infty$ for the closed-loop system composed of plant (6.2)-(6.3) and controller (6.20)-(6.21), (6.23), is defined as the set of all initial states in Ξ for which the evolution of the system remains in Ξ for all allowable infinite reference and disturbance sequences [Pannocchia and Kerrigan, 2005], i.e.

$$\mathcal{O}_\infty = \{\mathbf{x}(0) \in \Xi \,|\, \boldsymbol{\varphi}(\mathbf{x}(0), \mathbf{z}_d(\cdot), \mathbf{w}(\cdot), k) \in \Xi, \\ \forall \mathbf{z}_d(\cdot) \subset \mathcal{Z}, \forall \mathbf{w}(\cdot) \subset \mathcal{W}, \forall (\mathbf{A}, \mathbf{B}) \in \Gamma, \forall k \in \mathbb{N}^+\}. \tag{6.39}$$

Since Ξ is given by a finite number of affine inequality constraints, $\mathcal{O}_\infty$ is computed by efficient linear programming (LP) methods like the recursive Algorithm 6.2 proposed in [Kolmanovsky and Gilbert, 1998].

Note that if, in the construction of Ξ, an augmented form of the state vector is used, then one can define $\mathtt{X}_0$ as the set of states belonging to the plant state space, $\mathtt{X}$, for which, for all $\mathbf{x} \in \mathtt{X}_0$ there exists a corresponding augmented state that belongs to $\mathcal{O}_\infty$. $\mathtt{X}_0$ can be computed explicitly: since $\mathcal{O}_\infty$ is a polyhedron, then the set $\mathtt{X}_0$ is computed as the projection of $\mathcal{O}_\infty$ onto the plant state space, i.e. $\mathtt{X}_0 = Proj_{\mathtt{X}}(\mathcal{O}_\infty)$.

6.3.6 Invariant Set for the Proposed Controller

As explained in Subsection 6.3.5, the robust invariant set for the closed-loop system must be searched inside of Ξ. In other words, to find a robust invariant set for the closed-loop system, first the search for the constraint-admissible set must be

performed. This initial constraint-admissible set, i.e. Ξ, is obtained just after both the constrains on $\mathbf{u}$, and $\mathbf{x}$ are considered. By finding an invariant set, it is shown that the closed-loop system is asymptotically ultimately bounded, where for each initial condition inside of a bounded invariant set, the LPV system will converge asymptotically to this bounded set (Definition 6.3.2).

According to the controller proposed in Subsection 6.3.3, for the discrete-time uncertain system (6.2)-(6.3), the command signal $\mathbf{u}(k)$ is given by (6.23). Then, the closed-loop state equation is given by

$$\mathbf{x}(k+1) = (\mathbf{A} + \mathbf{B}\bar{\mathbf{A}}_1)\mathbf{x}(k) + \mathbf{B}\bar{\mathbf{A}}_2\mathbf{z}_d(k+1) + \mathbf{B}\bar{\mathbf{A}}_3\mathbf{z}_d(k) + \mathbf{B}\bar{\mathbf{A}}_4\bar{\mathbf{w}}(k-1) + \mathbf{w}(k). \tag{6.40}$$

Recall that the $\bar{\mathbf{w}}(k-1)$ computed in (6.20) meets $\bar{\mathbf{w}}(k-1) \in \mathcal{W}$. The set of admissible system states that meet restrictions (6.5), (6.7), and that make the command signal of the proposed controller (6.20)-(6.21), (6.23) meet restrictions (6.6), (6.8), for all admissible reference signals $\mathbf{z}_d(k) \in \mathcal{Z}$, subject to (6.10), and all possible disturbances $\mathbf{w}(k) \in \mathcal{W}$, subject to (6.9), is obtained by the constraint-admissible set Ξ (6.38). By using (6.23), the Ξ is given by

$$\Xi = \left\{ \mathbf{x} \in \mathbb{R}^n | \, \mathbf{x} \in \mathcal{X} \cap \bar{\mathbf{A}}_1\mathbf{x} \in \mathcal{F} \right\}, \tag{6.41}$$

where

$$\mathcal{F} = \mathcal{U} \sim \mathcal{L}, \tag{6.42}$$

$$\mathcal{L} = (\bar{\mathbf{A}}_2 * \mathcal{Z}) \oplus (\bar{\mathbf{A}}_3 * \mathcal{Z}) \oplus (\bar{\mathbf{A}}_4 * \mathcal{W}). \tag{6.43}$$

The initial constraint-admissible set Ξ is obtained from the consideration and enforcement of the constrains on both $\mathbf{u}(k)$, and $\mathbf{x}(k)$. In (6.41), the Ξ set is computed by applying the intersection operator between these two constraints. The second operand on this equation (i.e. $\bar{\mathbf{A}}_1\mathbf{x} \in \mathcal{F}$) represents the enforcement of the constraints that the $\mathbf{u}(k)$, $\mathbf{z}_d(k)$, and $\mathbf{w}(k)$ are assumed to meet. Therefore, before determining Ξ through (6.41), equations (6.42)-(6.43) should be used to determine $\mathcal{F}$ which in turn depends on the allowable operating region of $\mathbf{u}(k)$ (i.e. depends on $\mathcal{U}$) subject to the allowable operating regions for the setpoints and disturbance (i.e. subject to $\mathbf{z}_d(k) \in \mathcal{Z}$ and $\mathbf{w}(k) \in \mathcal{W}$). Once this region $\mathcal{F}$ is calculated by

(6.42)-(6.43), then it is applied in (6.41).

The Ξ in (6.41) is computed by solving a finite number of affine inequalities [Kolmanovsky and Gilbert, 1998]. The robustly positively invariant set $\mathcal{O}_\infty$ for the closed-loop system (6.40) is then represented by

$$\mathcal{O}_\infty = \{\mathbf{x}(k) \in \Xi \,|\, \mathbf{x}(k+1) \in \Xi, \\ \forall \mathbf{z}_d(k+1) \in \mathcal{Z}, \forall \mathbf{z}_d(k) \in \mathcal{Z}, \forall \mathbf{w}(\cdot) \subset \mathcal{W}, \forall (\mathbf{A}, \mathbf{B}) \in \Gamma, \forall k \in \mathbb{N}^+\}. \quad (6.44)$$

6.4 Robust Control Design Based on PI-Type Synergetic Control Theory Structure

Consider a class of nonlinear state dependent discrete time varying (NSDDTV) dynamical systems with disturbance, represented by the following equations:

$$\mathbf{x}(k+1) = \mathbf{A}(\mathbf{x}, k)\mathbf{x}(k) + \mathbf{B}(\mathbf{x}, k)\mathbf{u}(k) + \mathbf{w}(k), \quad (6.45)$$

$$\mathbf{y}(k) = \mathbf{C}(\mathbf{x}, k)\mathbf{x}(k), \quad (6.46)$$

where $\mathbf{x}(k) = [x_1(k), \ldots, x_n(k)]^T \in \mathbb{R}^n$ is the plant state which is assumed to be measured, $\mathbf{u}(k) = [u_1(k), \ldots, u_m(k)]^T \in \mathbb{R}^m$ is the control input, $\mathbf{w}(k) = [w_1(k), \ldots, w_n(k)]^T \in \mathbb{R}^n$ is a bounded, and unmeasured disturbance, $\mathbf{y}(k) = [y_1(k), \ldots, y_p(k)]^T \in \mathbb{R}^p$ is the output variable, and $\mathbf{C}(\mathbf{x}, k) = [c_{ij}(\mathbf{x}, k)] \in \mathbb{R}^{p \times n}$ is a non-negative matrix (i.e. $c_{ij}(\mathbf{x}, k) \geqslant 0$ for all $i = 1, \ldots, p$, $j = 1, \ldots, n$). $\mathbf{y}(k)$ is also considered the controlled variable, i.e. the variable to be controlled to a given desired setpoint sequence $\mathbf{z}_d(k)$. The $\mathbf{A}(\mathbf{x}, k)$, $\mathbf{B}(\mathbf{x}, k)$, and $\mathbf{C}(\mathbf{x}, k+1)$ matrices can be time-varying, and dependent on the state $\mathbf{x}(k)$, but to simplify the notation, in the sequel they will be simply denoted by $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, respectively.

It is assumed that the triplet $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ of the nominal system is both controllable and observable with the matrices $\mathbf{B}$ and $\mathbf{C}$ being of full rank. In addition matrix $\mathbf{CB}$ is supposed to be invertible. It is assumed that, at each time instant k, the disturbances $\mathbf{w}(k+p)$ are known for $p < 0$, but are unknown for $p \geqslant 0$. It is also assumed that Assumption 6.3.3 holds. Similarly to what was noted in Subsection 6.3.1 regarding system (6.2)-(6.3), note that under fairly general conditions a state space model (6.45)-(6.46) can be transformed into an observability, observer, or

observable canonical form/realization where $\mathbf{C}(k)$ is non-negative [Sinha and Rózsa, 1976; Antsaklis and Michel, 2007; Tóth, 2010].

This section presents a novel robust controller for nonlinear state dependent discrete time varying (NSDDTV) systems of the form (6.45)-(6.46), such that, for any allowable (i.e. bounded) disturbance and varying setpoints sequence, the controller accomplishes the goal of asymptotically driving the system towards a bounded macro-variable manifold based on output error and containing the origin, and then maintaining it there.

6.4.1 Proposed Controller for NSDDTV Systems

In this section, the discrete time controller based on synergetic control theory (SCT) is designed in the presence of unmeasured bounded additive disturbances on nonlinear state dependent discrete time varying (NSDDTV) systems (6.45)-(6.46) [Rastegar *et al.*, 2017a]. First, let the tracking error be defined as $\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{z}_d(k)$, where $\mathbf{z}_d(k)$ contains the setpoints, and define a proportional integral (PI) type macro-variable, as follows:

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{z}_d(k), \tag{6.47}$$

$$\boldsymbol{\psi}(k) = \mathbf{K}_P\mathbf{e}(k) + \mathbf{K}_I\boldsymbol{\zeta}(k), \tag{6.48}$$

$$\boldsymbol{\psi}(k+1) = \mathbf{K}_P\mathbf{e}(k+1) + \mathbf{K}_I\boldsymbol{\zeta}(k+1), \tag{6.49}$$

where $\boldsymbol{\zeta}(k)$ is an integral-type of the error which is defined as $\boldsymbol{\zeta}(k) = \mathbf{e}(k)+\boldsymbol{\zeta}(k-1)$, and $\mathbf{K}_P$ and $\mathbf{K}_I$ are positive proportional and integral gains, respectively.

Replacing (6.48) and (6.49) into the discrete-time version of (6.12) (see Subsection 6.3.2), discretized by the forward differences method, yields

$$T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{e}(k+1) + T\mathbf{K}_I\boldsymbol{\zeta}(k) - (T - T_s)\boldsymbol{\psi}(k) = 0. \tag{6.50}$$

Eq. (6.50) expresses the dynamic error of the system, where T_s is the sampling time period.

Using (6.45), and (6.47), the error dynamics equation (6.50) can be rewritten as:

$$\begin{aligned} &T(\mathbf{K}_P + \mathbf{K}_I)[\mathbf{CAx}(k) + \mathbf{CBu}(k) + \mathbf{Cw}(k) - \mathbf{z}_d(k+1)] \\ &+ T\mathbf{K}_I\boldsymbol{\zeta}(k) - (T - T_s)\boldsymbol{\psi}(k) = 0. \end{aligned} \tag{6.51}$$

From (6.46), (6.47), and (6.51), the input signal is obtained as

$$\begin{aligned}\mathbf{u}(k) = &- \left[T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{CB}\right]^{-1} \left[-q\boldsymbol{\psi}(k) + T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{CA}\mathbf{x}(k)\right. \\ &\left. + T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{C}\mathbf{w}(k) - T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{z}_d(k+1) + T\mathbf{K}_I\boldsymbol{\zeta}(k)\right]. \end{aligned} \tag{6.52}$$

where $q = T - T_s$, and it is assumed that $\det(\mathbf{CB}) \neq 0$. In addition, similarly to Theorem 6.3.1 in Subsection 6.3.4, it is assumed that $T > T_s/2 > 0$, and thus $-T_s < q < 0$.

In Subsection 6.3.3 some points were remarked regarding the mathematical implications of the assumption that $\det(\mathbf{CB}) \neq 0$. Here, an additional point is remarked regarding this assumption. Define the *certain LTV system* to be the LTV system (6.45)-(6.46) with $\mathbf{w}(k)$ being zero or known at time instant k, for all k. Then, generalizing the analysis in [Åström and Wittenmark, 1997, Sec. 3.4], $\mathbf{y}(k_1) = \mathbf{C}(k_1)\boldsymbol{\Phi}(k_0, k_1)\mathbf{x}(k_0) + \mathbf{W}_c(k_0, k_1)\mathbf{U}$, where the controlability matrix $\mathbf{W}_c(k_0, k_1)$ is defined such that $\mathbf{W}_c(k_0, k_1)\mathbf{U} = \mathbf{C}(k_1)\sum_{k=k_0}^{k_1-1}\boldsymbol{\Phi}(k+1, k_1)\mathbf{B}(k)\mathbf{u}(k)$, $\mathbf{U} = [\mathbf{u}(k_0), \ldots, \mathbf{u}(k_1 - 1)]^T$, $\boldsymbol{\Phi}(k_0, k_1) = \prod_{k=k_0}^{k_1-1}\mathbf{A}(k)$, except that $\boldsymbol{\Phi}(k_0, k_0) = \mathbf{I}$ is the identity matrix, for all k_1, k_0 such that $k_1 \geqslant k_0$. Then, $\det(\mathbf{CB}) \neq 0$ is a sufficient condition for $\mathbf{W}_c(k_0, k_1)$ to be full rank, which in turn is an equivalent condition for such certain LTV system to be output reachable (i.e. to be possible for the output $\mathbf{y}(k_1)$ to be transferred to any value by the adequate selection of actuation commands $\mathbf{U}$ to the process). Moreover, in such certain LTV system, if $\det(\mathbf{CB}) \neq 0$ then in theory it is possible to place $\mathbf{y}(k+1)$ arbitrarily by choosing the right value of $\mathbf{u}(k)$. However, in this chapter the disturbance $\mathbf{w}(k)$ is not known at sampling time k, which implies that it is not possible that one can place $\mathbf{y}(k+1)$ arbitrarily by choosing the right value of $\mathbf{u}(k)$. Therefore, condition of $\det(\mathbf{CB}) \neq 0$ should not be interpreted as a reachability condition for system (6.45)-(6.46), as it can be realized for the certain LTV system.

The command law (6.52) needs to have an explicit form. By this reason, and assuming for the sake of motivation that $\mathbf{w}(k)$ is smooth so that the change of $\mathbf{w}(k)$ in one sampling interval is small, $\mathbf{w}(k) \approx \mathbf{w}(k-1)$, then, similarly to Subsection 6.3.3, the unknown bounded disturbance $\mathbf{w}(k)$ is replaced by $\mathbf{w}(k-1)$ which is the one-step delayed estimation of $\mathbf{w}(k)$ [Xu and Li, 2012] obtained from (6.45).

Specifically,

$$\mathbf{w}(k-1) = \mathbf{x}(k) - \mathbf{A}(k-1)\mathbf{x}(k-1) - \mathbf{B}(k-1)\mathbf{u}(k-1). \tag{6.53}$$

By replacing the unknown bounded $\mathbf{w}(k)$ with (6.53), then (6.52) is transformed into:

$$\begin{aligned}\mathbf{u}(k) = &- \left[T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{CB})\right]^{-1} \left[-q\boldsymbol{\psi}(k) + T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{CAx}(k)\right. \\ &\left. + T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{Cw}(k-1) - T(\mathbf{K}_P + \mathbf{K}_I)\mathbf{z}_d(k+1) + T\mathbf{K}_I\boldsymbol{\zeta}(k)\right], \end{aligned} \tag{6.54}$$

where $\mathbf{w}(k-1)$ is given by (6.53).

Define the asymptotic stability of the closed-loop system composed of plant (6.45)-(6.46) and controller (6.54) as the property of the system where the error trajectory to moves towards a bounded SCT macro-variable manifold starting from any initial state, and then remains in such manifold. The asymptotic stability can be achieved by choosing the controller parameters T, T_s, $\mathbf{K}_P$, and $\mathbf{K}_I$. Again, note that in situations where it is not possible to predict $\mathbf{z}_d(k+1)$, then the command signal $\mathbf{u}(k)$ in (6.52), (6.54), can be approximated/estimated, by using $\mathbf{z}_d(k+1) \approx 2\mathbf{z}_d(k) - \mathbf{z}_d(k-1)$ to extrapolate for the next future reference signal, $\mathbf{z}_d(k+1)$.

The controller proposed in this section is termed robust discrete SCT controller of proportional-integral type (RDSCTC-PI).

6.4.2 Robust Closed-Loop Stability

The stability analysis for the proposed controller based on a PI-type of SCT macro-variable follows a same procedure as for the RDSCTC (Subsection 6.3.4). The strategy used to design the control law for the process, is that the SCT convergence law in (6.12) must guarantee the following desired attributes: (1) starting from any initial state, the output error is moved towards a bounded macro-variable manifold based on error and containing the origin, where such manifold is also called the robust macro-variable manifold band (*RMMB*); (2) After the trajectory has reached the *RMMB*, then in each subsequent time step, the trajectory is confined to remain

inside the *RMMB*, where

$$RMMB = \left\{ \mathbf{x} \in \mathbb{R}^n | \ |\boldsymbol{\psi}(\mathbf{x}, \mathbf{z}_d)| \leqslant M_\psi = \boldsymbol{\Omega}\mathbf{C}(\mathbf{w}_{max} - \mathbf{w}_{min}) \frac{T}{2T - T_s} \right\}, \tag{6.55}$$

and $\boldsymbol{\Omega} = \mathbf{K}_P + \mathbf{K}_I$. $\mathbf{w}_{min} \in \mathbb{R}^n$ and $\mathbf{w}_{max} \in \mathbb{R}^n$ are the row-wise minimum and maximum of $\mathbf{w}(k)$, respectively.

The control law that enforces the SCT convergence law in Eq. (6.12) is given by (6.54), and is rewritten as follows:

$$\begin{aligned} \mathbf{u}(k) = &- (T\boldsymbol{\Omega}\mathbf{C}\mathbf{B})^{-1} \left[-q\boldsymbol{\psi}(k) + T\boldsymbol{\Omega}\mathbf{C}\mathbf{A}\mathbf{x}(k) \right. \\ &\left. + T\boldsymbol{\Omega}\mathbf{C}\mathbf{w}(k-1) - T\boldsymbol{\Omega}\mathbf{z}_d(k+1) + T\mathbf{K}_I\boldsymbol{\zeta}(k) \right]. \end{aligned} \tag{6.56}$$

Theorem 6.4.1. *Consider the system (6.45)-(6.46) with control law (6.56). Suppose the reference/desired output variables are given by $\mathbf{z}_d(k)$, and the dynamic error is defined as $\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{z}_d(k)$. Also assume that $T < T_s < 2T$, and $\det(\mathbf{C}\mathbf{B}) \neq 0$. Then, the closed-loop system (6.45)-(6.46), (6.56), in the presence of unknown bounded disturbance $\mathbf{w}(k)$, is stable, and using the control law (6.56) the error trajectory asymptotically converges to the* RMMB *(6.55). After the trajectory has reached the* RMMB*, then in each subsequent time step, the trajectory is confined to remain inside the* RMMB*.*

Proof. Using (6.49), (6.45), and (6.56), start by noting that

$$\begin{aligned} \boldsymbol{\psi}(k+1) &= \boldsymbol{\Omega}\mathbf{e}(k+1) + \mathbf{K}_I\boldsymbol{\zeta}(k) \\ &= \boldsymbol{\Omega}\left[\mathbf{C}\mathbf{x}(k+1) - \mathbf{z}_d(k+1)\right] + \mathbf{K}_I\boldsymbol{\zeta}(k) \\ &= \boldsymbol{\Omega}\left[\mathbf{C}(\mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{w}(k)) - \mathbf{z}_d(k+1)\right] + K_I\boldsymbol{\zeta}(k) \\ &= \boldsymbol{\Omega}\mathbf{C}\mathbf{A}\mathbf{x}(k) - \boldsymbol{\Omega}\mathbf{C}\mathbf{B}\left(T\boldsymbol{\Omega}\mathbf{C}\mathbf{B}\right)^{-1}\left[q\boldsymbol{\psi}(k) + T\boldsymbol{\Omega}\mathbf{C}\mathbf{A}\mathbf{x}(k)\right. \\ &\quad \left. + T\boldsymbol{\Omega}\mathbf{C}\mathbf{w}(k-1) - T\boldsymbol{\Omega}\mathbf{z}_d(k+1) + T\mathbf{K}_I\boldsymbol{\zeta}(k)\right] \\ &\quad + \boldsymbol{\Omega}\mathbf{C}\mathbf{w}(k) - \boldsymbol{\Omega}\mathbf{z}_d(k+1) + \mathbf{K}_I\zeta(k), \\ &= \boldsymbol{\Omega}\mathbf{C}\mathbf{A}\mathbf{x}(k) - \boldsymbol{\Omega}\mathbf{C}\mathbf{A}\mathbf{x}(k) - \boldsymbol{\Omega}\mathbf{C}\mathbf{w}(k-1) - T\mathbf{K}_I\boldsymbol{\zeta}(k) \\ &\quad + \boldsymbol{\Omega}\mathbf{z}_d(k+1) - \left(-1 + \frac{T_s}{T}\right)\boldsymbol{\psi}(k) + \boldsymbol{\Omega}\mathbf{C}\mathbf{w}(k) \\ &\quad - \boldsymbol{\Omega}\mathbf{z}_d(k+1) + T\mathbf{K}_I\boldsymbol{\zeta}(k) \\ &= \left(1 - \frac{T_s}{T}\right)\boldsymbol{\psi}(k) + \boldsymbol{\Omega}\mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)) \end{aligned}$$

$$= qT^{-1}\boldsymbol{\psi}(k) + \boldsymbol{\Omega}\mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1))$$
$$= \frac{T - T_s}{T}\boldsymbol{\psi}(k) + \boldsymbol{\Omega}\mathbf{C}(\mathbf{w}(k) - \mathbf{w}(k-1)). \quad (6.57)$$

By comparing (6.25), and (6.27) with (6.55), and (6.57), respectively, it can be realized that the proof of Theorem 6.4.1 regarding stability analysis/properties of the proposed NSDDTV controller based on a PI-type SCT macro-variable can be performed similarly to the proof of Theorem 6.3.1 (Subsection 6.3.4) regarding stability analysis/properties of the RDSCTC controller. In fact, the only difference is that matrix $\mathbf{C}$ in (6.25), and (6.27), is substituted by $\boldsymbol{\Omega}\mathbf{C}$ in (6.55) and (6.57). Otherwise, the proof of Theorem 6.4.1 follows the same procedure as the proof of Theorem 6.3.1. □

6.5 Experimental Results

This section, presents three examples that illustrate the implementation of the proposed RDSCTC, and of the proposed RDSCTC-PI. The performance and effectiveness of the RDSCTC is demonstrated through an example of reactor temperature control on a nonisothermal Continuous Stirred Tank Reactor (CSTR) plant. The performance of the RDSCTC-PI is presented through the control of the acetate concentration $A(t)$ in the Escherichia Coli (E. Coli) process bioreactor. Additionally, to better demonstrate the efficiency of RDSCTC, and RDSCTC-PI, the closed-loop simulation results obtained by RDSCTC, and RDSCTC-PI are demonstrated and compared with the performance of the RMPC [Pannocchia, 2004], and with the standard discrete-time sliding mode controller (SDSMC) [Eun *et al.*, 1999]. Finally, the performances of all controllers proposed in this thesis, i.e. the appplication of the AFGPC (Method 1), AFGPC (Method 2), RDSCTC, and RDSCTC-PI on the real-world experimental setup composed of two DC motors (Subsetion 4.4.4) are illustrated and compared. All the simulations and computations were performed on an Intel(R) core (TM) i7-2600, CPU 3.4GHz, with 512 MB Cache, and total memory of 8 GB. The constraint admissible sets Ξ, and the robustly positively invariant sets $\mathcal{O}_\infty$ were computed using the Multi-Parametric Toolbox [Herceg *et al.*, 2013], and the Invariant Set Toolbox [Kerrigan, 2005].

Table 6.1: Variables of the continuous stirred tank reactor (CSTR) [Wan and Kothare, 2002].

Variables description	Value
C_A - Product concentration	$0.265\,[\text{mol/l}]$
T_c - Reactor temperature	$394\,[\text{K}]$
q_c - Coolant flow rate	$250\,[\text{l/min}]$
q - Process flow rate	$1 \times 10^3\,[\text{l/min}]$
C_{A0} - Feed concentration	$1\,[\text{mol/l}]$
T_o - Feed temperature	$365\,[\text{K}]$
T_{c0} - Inlet coolant temperature	$365\,[\text{K}]$
V - CSTR volume	$1 \times 10^3\,[\text{l}]$
hA - Heat transfer term	$7 \times 10^5\,[\text{cal/min/K}]$
k_0 - Reaction rate constant	$1 \times 10^9 \leqslant k_0 \leqslant 5 \times 10^9\,[\text{min}^{-1}]$
E/R - Activation energy term	$8330.1\,[\text{K}]$
$-\Delta H$ - Heat of reaction	$1 \times 10^7 \leqslant -\Delta H \leqslant 5 \times 10^7\,[\text{cal/mol}]$
ρ, ρ_c - Liquid densities	$1 \times 10^3\,[\text{g/l}]$
C_p, C_{pc} - Specific heats	$1\,[\text{cal/g/K}]$

6.5.1 Control of a Simulated CSTR Plant

A nonisothermal Continuous Stirred Tank Reactor (CSTR) is a highly nonlinear process which is very common in chemical and petrochemical plants. It is frequently used to show the efficiency of control designs, including for LPV systems, e.g. [Wan and Kothare, 2002], [Pannocchia, 2004]. In the process, a single irreversible exothermic reaction is assumed to occur in the reactor. The nonisothermal CSTR for an exothermic irreversible reaction is described by the following dynamic model based on a component balance and on an energy balance [Morningred *et al.*, 1992]:

$$\frac{\partial C_A(t+d_c)}{\partial t} = \frac{q(t)}{V}\left(C_{A0}(t) - C_A(t+d_c)\right) - k_0 C_A(t+d_c)\exp\left(-\frac{E}{RT_c(t)}\right), \quad (6.58)$$

$$\begin{aligned}\frac{\partial T_c(t)}{\partial t} = {} & \frac{q(t)}{V}\left(T_0(t) - T_c(t)\right) - \frac{(-\Delta H)k_0 C_A(t+d_c)}{\rho C_p}\exp\left(-\frac{E}{RT_c(t)}\right) \\ & + \frac{\rho_c C_{pc}}{\rho C_p V} q_c(t)\left[1 - \exp\left(\frac{-hA}{q_c(t)\rho_c C_{pc}}\right)\right]\left(T_{c0}(t) - T_c(t)\right). \end{aligned} \quad (6.59)$$

The objective is to control the reactor temperature $T_c(t)$ asymptotically by manipulation of the coolant flow rate $q_c(t)$. It is assumed that the CSTR plant is exposed to polytopic model uncertainties, state and input constraints, and plant

disturbances. The plant variables and the respective nominal values are described in Table 6.1.

A linearization operation is performed on the continuous-time model (6.58)-(6.59), yielding a linear model which is given in terms of perturbation/deviation variables. At a steady operating point of $C_A = 0.265\,[\text{mol/l}]$ and $T_c = 394\,[K]$, and under uncertainty in the parameters k_0 and $-\Delta H$, the linear model is discretized using a sampling period of $T_s = 0.15\,[\text{min}]$, which results in the following discrete-time linear model [Wan and Kothare, 2002]:

$$\mathbf{x}(k+1) = \begin{bmatrix} 0.85 - 0.0986\alpha(k) & -0.0014\alpha(k) \\ 0.9864\alpha(k)\beta(k) & 0.0487 + 0.01403\alpha(k)\beta(k) \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 0 \\ -0.912 \end{bmatrix} u(k) + \mathbf{w}(k), \tag{6.60}$$

$$y(k) = \begin{bmatrix} 0 & 1 \end{bmatrix} \mathbf{x}(k), \tag{6.61}$$

where $\mathbf{w}(k) = [w_1(k), w_2(k)]^T$ represents a disturbance, and the following parameter uncertainties are assumed: $1 \leqslant \alpha(k) = k_0/10^9 \leqslant 5$ and $1 \leqslant \beta(k) = -\Delta H/10^7 \leqslant 5$. In the experiments $\alpha(k)$ and $\beta(k)$ are implemented as random variables uniformly distributed within the intervals defined by these bounds. The experiments used the linearized model (6.60)-(6.61). Notice that in this example model (6.60)-(6.61), matrix $\mathbf{B}(k)$ has no uncertainty, so that $\mathbf{B}(k)$ will not be considered in the model uncertainty. The following constraints on the plant states and input and on the admissible setpoints and possible disturbances are considered:

$$\begin{aligned} &-1.5 \leqslant u(k) \leqslant 1.5, && -1 \leqslant z_d(k) \leqslant 1, \\ &-\begin{bmatrix} 0.5 \\ 5 \end{bmatrix} \leqslant \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \leqslant \begin{bmatrix} 0.5 \\ 5 \end{bmatrix}, && -\begin{bmatrix} 0.05 \\ 0.25 \end{bmatrix} \leqslant \begin{bmatrix} w_1(k) \\ w_2(k) \end{bmatrix} \leqslant \begin{bmatrix} 0.05 \\ 0.25 \end{bmatrix}, \end{aligned} \tag{6.62}$$

where $\mathbf{z}_d(k) = z_d(k)$ is the set-point.

Equation (6.60) contains uncertain parameters $\alpha(k)$, and $\beta(k)$, each of them bounded between two values. Thus, the polytopic uncertain set has four vertices $\Omega = Co\,\{\mathbf{A}_1^v, \mathbf{A}_2^v, \mathbf{A}_3^v, \mathbf{A}_4^v\}$. For the RDSCTC, the nominal model $\bar{\mathbf{A}}$ for the CSTR is considered to be the average of the four vertices, and parameters $T = 0.085\,[\text{min}]$, and $T_s = 0.15\,[\text{min}]$ were chosen. The applied $\mathbf{w}(k)$ is a random noise with uniform distribution within the $\mathbf{w}(k)$ bounds defined in (6.62), except for $7.5\,[\text{hours}] \leqslant t \leqslant$

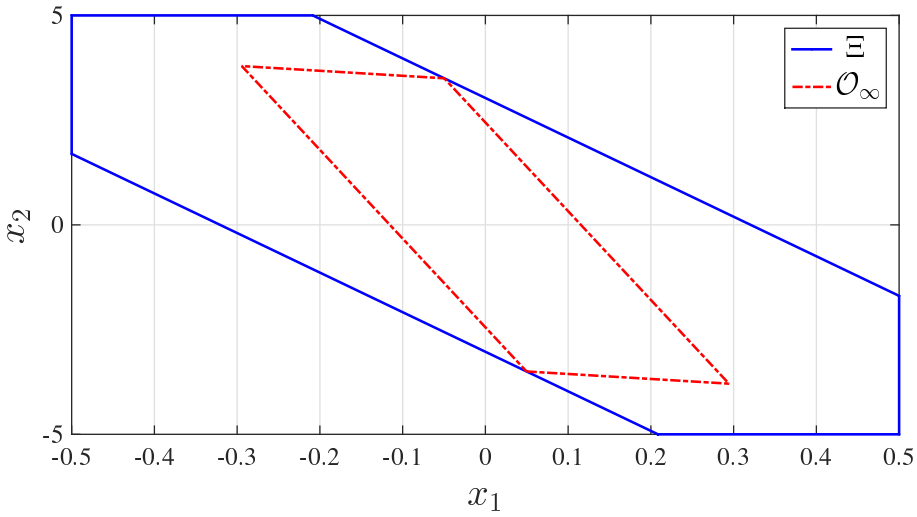


Figure 6.6: Illustration of the constraint-admissible set Ξ, and the robust control invariant set $\mathcal{O}_\infty$ for the RDSCTC on the CSTR plant.

9 [hours], where $t = kT_s$, and $w_2(k) = w_e(k) + w_s(k)$, $w_e(k) = 0.15$, and $w_s(k)$ is uniformly distributed within the bounds $-0.1 \leqslant w_s(k) \leqslant 0.1$.

Figure 6.6 presents the constraint-admissible set Ξ (6.38), (6.41), and the robustly positively invariant set $\mathcal{O}_\infty$ (6.39), (6.44) of the CSTR plant. To demonstrate the feasibility of RDSCTC, three initial state points are chosen randomly inside of the robustly positively invariant set $\mathcal{O}_\infty$, namely $\mathbf{x}_A = [-0.2081, 3.882]^T$, $\mathbf{x}_B = [-0.0712, 3.611]^T$, and $\mathbf{x}_C = [0.1735, -3.836]^T$, while the controlled variable setpoint $z_d(t)$, is changed during the simulation as follows:

$$z_d(t) = \begin{cases} 1, & 0 < t \leqslant 3.33\,[\text{hours}], \\ -1, & 3.33\,[\text{hours}] < t \leqslant 6.67\,[\text{hours}], \\ 0, & 6.67\,[\text{hours}] < t \leqslant 10\,[\text{hours}]. \end{cases} \tag{6.63}$$

The state trajectories obtained with the different initial states $\mathbf{x}_A$, $\mathbf{x}_B$, and $\mathbf{x}_C$, and the result obtained by the proposed RDSCTC on the nonlinear simulated CSTR plant (6.58)-(6.59) after starting from the initial state $\mathbf{x}_D = [0.10, -3.65]^T$, are presented in Figure 6.7.

Figure 6.8 presents the closed-loop simulation results obtained by the proposed RDSCTC on the discrete-time linearized CSTR plant (6.60)-(6.61) after starting

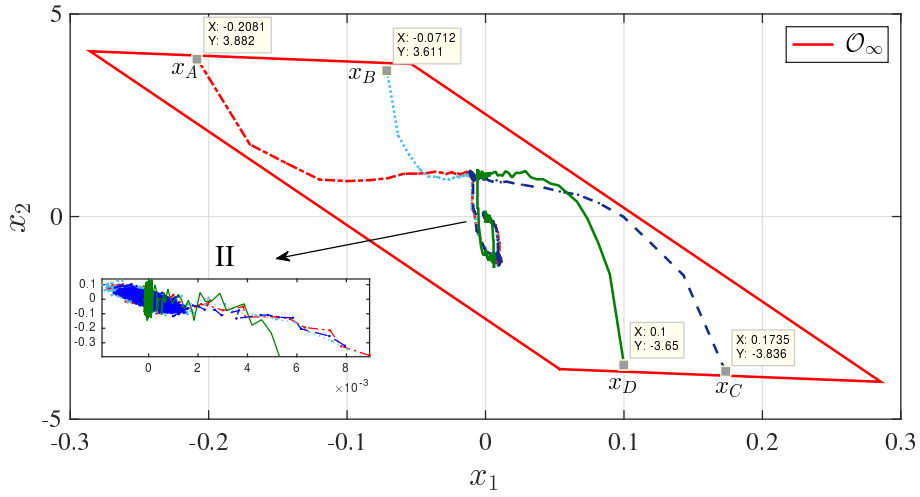
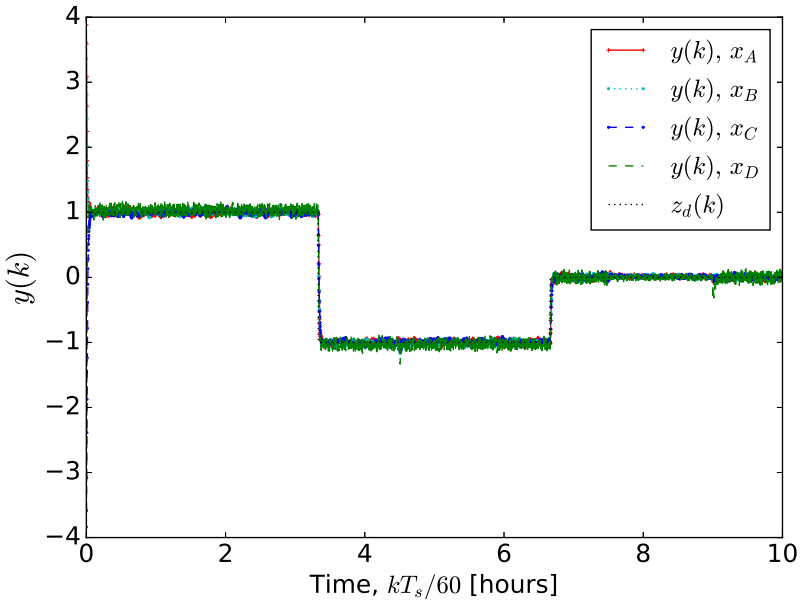


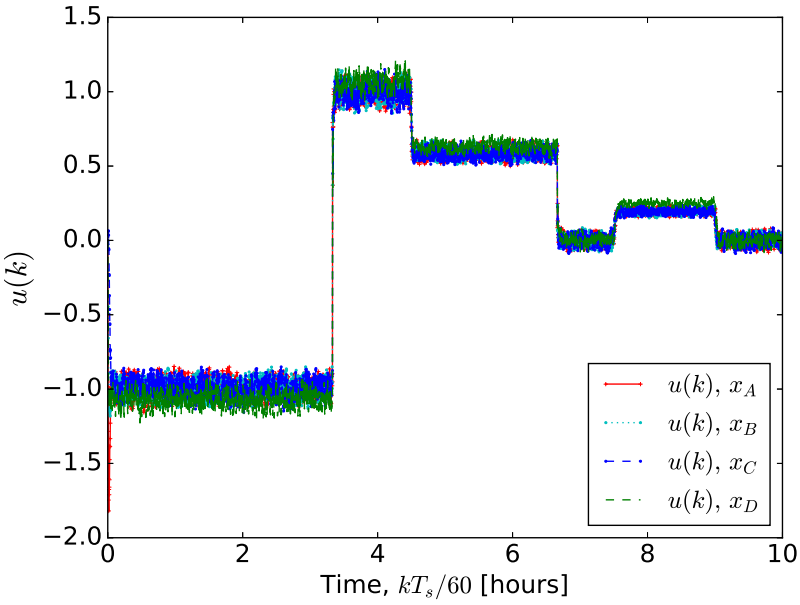
Figure 6.7: The robust control invariant set $\mathcal{O}_\infty$ in the $\mathbf{x}$ space, and the state trajectories for different initial states of $\mathbf{x}_A$, $\mathbf{x}_B$, $\mathbf{x}_C$, and $\mathbf{x}_D$ for RDSCTC on the CSTR plant and the reference signal given by (6.63).

from initial states $\mathbf{x}_A$, $\mathbf{x}_B$, and $\mathbf{x}_C$, and the result obtained by the RDSCTC on the nonlinear simulated CSTR plant (6.58)-(6.59) after starting from $\mathbf{x}_D$. During the experiment, two different models have been used to implement the plant. For $t \leqslant 4.5\,[\text{hours}]$, the nominal plant model $\bar{\mathbf{A}}$ was the average of the four vertices, i.e. $\mu(1) = \mu(2) = \mu(3) = \mu(4) = \frac{1}{4}$, where $\mu(j)$ $(j = 1, 2, 3, 4)$ have the meaning defined in Subsection 6.3.1. For $t > 4.5\,[\text{hours}]$ an unexpected change was inserted into the nominal plant model: the new plant model is obtained by $\mu(1) = \frac{1}{8}$, $\mu(2) = \frac{4}{8}$, $\mu(3) = \frac{2}{8}$, $\mu(4) = \frac{1}{8}$. Note that both nominal models that are in effect before or after $t = 4.5\,[\text{hours}]$ are inside Γ (6.4). The results in the Figure 6.8 reveal that, as it was expected, the RDSCTC enforces the controlled variable to track the setpoints despite of the presence of unmeasured disturbances and uncertainty. When the setpoints are changed, the controllers drive the controlled variable to the new setpoints.

The results of the $\psi(k) = e(k) = y(k) - z_d(k)$ functions for the three different initial states are depicted in Figure 6.9. A value of $M_\psi = 2.125$ was obtained from (6.25). As can be seen in Figure 6.9(b), after the trajectory has entered the *RMMB*, then with every successive time step, the trajectory remains confined within the *RMMB*. The state trajectories manipulated by the RDSCTC were magnified around origin and mapped in part II inside Figure 6.7.

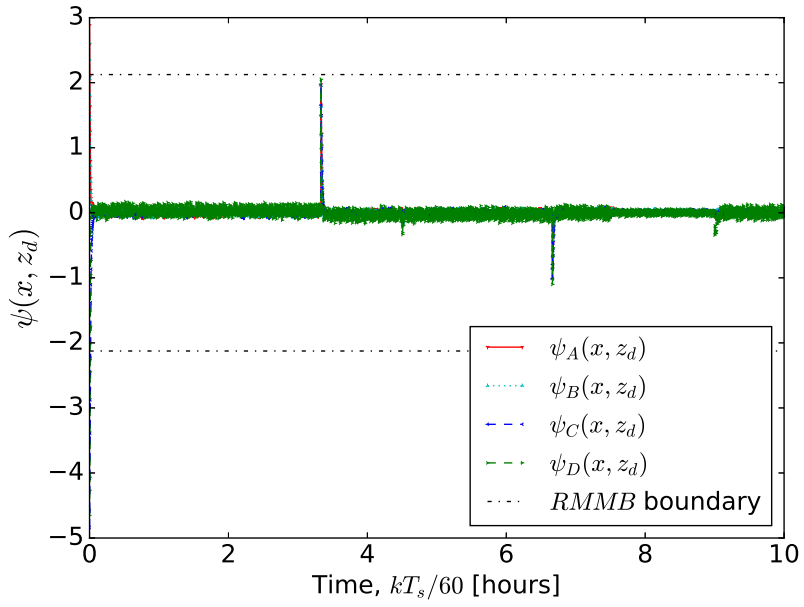


(a)

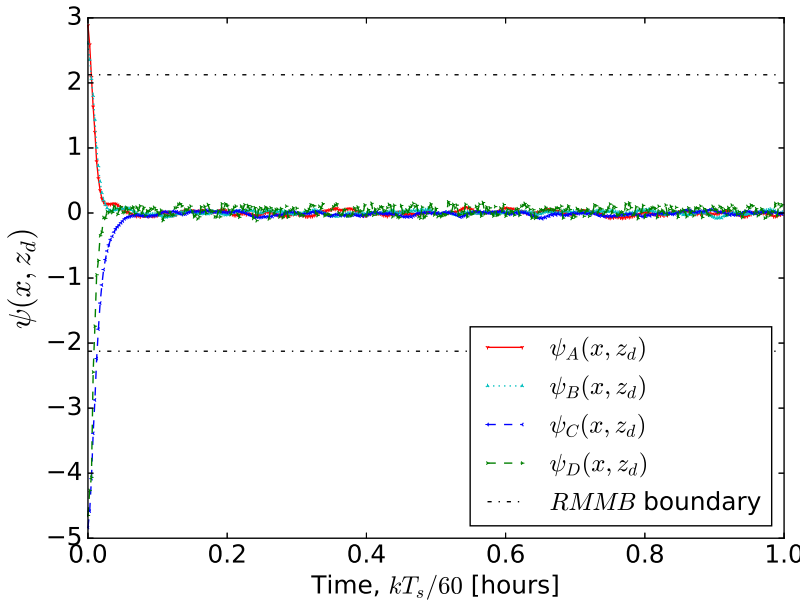


(b)

Figure 6.8: (a) Closed loop simulation results of the output $y(k)$ using the proposed RDSCTC for different initial states $\mathbf{x}_A$, $\mathbf{x}_B$, $\mathbf{x}_C$, and $\mathbf{x}_D$ on the CSTR plant; and (b) the respective applied command signals. The reference signal is given by (6.63).



(a)



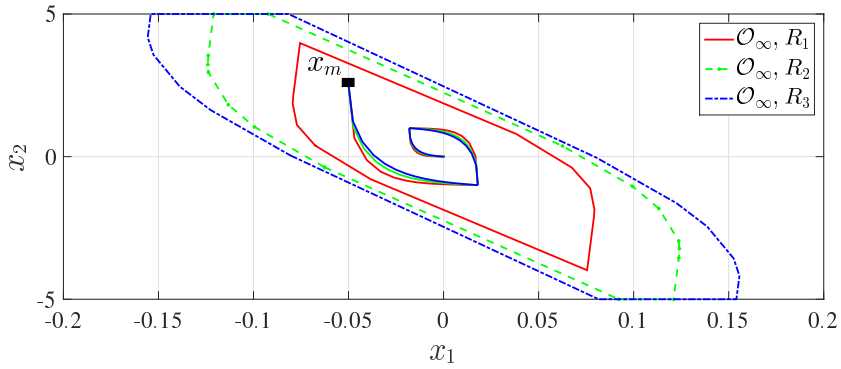
(b)

Figure 6.9: Results of the RDSCTC on the CSTR plant: (a) Illustration of the $\psi(k)$ function for the different initial states of $\mathbf{x}_A$, $\mathbf{x}_B$, $\mathbf{x}_C$, and $\mathbf{x}_D$ for the reference signal given by (6.63); and (b) Magnified plot of Figure 6.9(a) for the time period between 0 and 1 [hours].

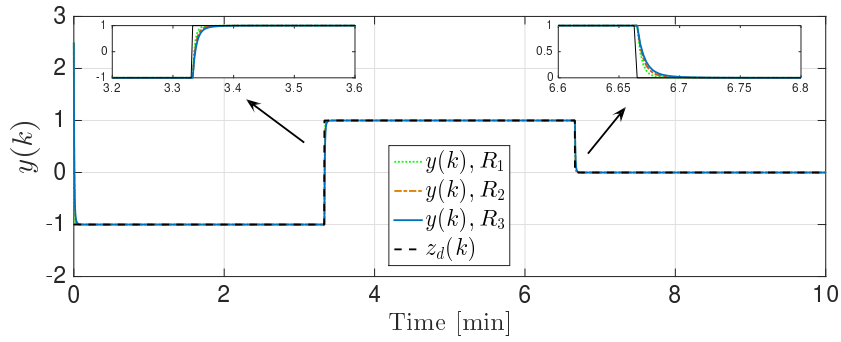
Parameters Sensitivity

In this section the sensitivity of the RDSCTC to its control parameters is investigated. For that purpose, the RDSCTC is run with different values of the SCT convergence rate T. The results obtained by the RDSCTC are compared with the work [Pannocchia, 2004]. As discussed in Section 6.1, [Pannocchia, 2004] presented a novel robust model predictive controller (RMPC) for LPV systems. Here, the experiments for both the RDSCTC and the RMPC used the same DT-LPV model of the CSTR plant as exemplified in [Pannocchia, 2004]. The RMPC was appointed to guarantee offset removal for setpoints different from the origin while respecting the process constraints. In the RMPC, the $\mathbf{R}$ is defined as a symmetric and positive definite matrix penalty factor for the actuation command in an infinite-horizon linear quadratic regulator (LQR) problem. An example with the same constraints, and setpoint as was defined in [Pannocchia, 2004] is defined for the RDSCTC. The RMPC performance for three values of $\mathbf{R} = R = R_1 = 0.05$, $\mathbf{R} = R = R_2 = 0.1$, and $\mathbf{R} = R = R_3 = 0.15$ is presented. In another part of the simulation study, the performance of the RDSCTC for three values of $T = T_1 = 3.8$, $T = T_2 = 4$, and $T = T_3 = 4.2$ is tested. The initial state value $\mathbf{x}_m = [-0.05, 2.5]^T$, and sampling time $T_s = 5\,[\mathrm{s}]$ were used. For the RMPC, $S = 1$, $N = 0$ (notice that, for the control horizon value of $N = 0$, the feasible region is the robustly invariant region), and $(A_m, B_m) = 0.6(A_1, B_1) + 0.4(A_2, B_2)$ were chosen. The results obtained by the RMPC and RDSCTC are depicted in Figures 6.10 and 6.11, respectively.

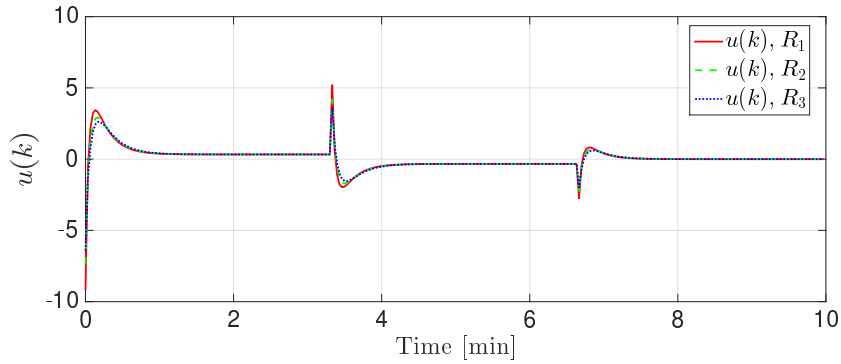
The performances of the time responses of RMPC and RDSCTC are comparable. However, three points are noticeable. First, the results show that comparing with RMPC, a larger invariant set region is ensured/obtained by the RDSCTC. Second, the RDSCTC showed less sensitivity to unexpected control parameter changes when compared to the RMPC. With a small variation in the value of R, the invariant set obtained by RMPC can shrink seriously. However, as can be seen in Figure 6.11(a), RDSCTC did not show such large sensitivity against changes in T. This is consistent with the fact that, as mentioned in Subsection 6.3.4, for the RDSCTC the condition of $T < T_s < 2T$ is sufficient to ensure the robustness of the control scheme in satisfying the control purposes. Third, in the RMPC, for prediction horizons larger that $N = 0$, the computational time is experimentally observed to grow much larger than the fixed computational time in RDSCTC. To show this, the



(a)



(b)



(c)

Figure 6.10: (a) Performance of the RMPC [Pannocchia, 2004] for different positive definite values of the variable R ($R = R_1, R_2, R_3$), to calculate the invariant set on the CSTR plant, where the other RMPC configurations are the same as the ones that were exemplified in [Pannocchia, 2004]; (b) and (c) the plant output and the command signal that resulted from the application of the RMPC controller, respectively.

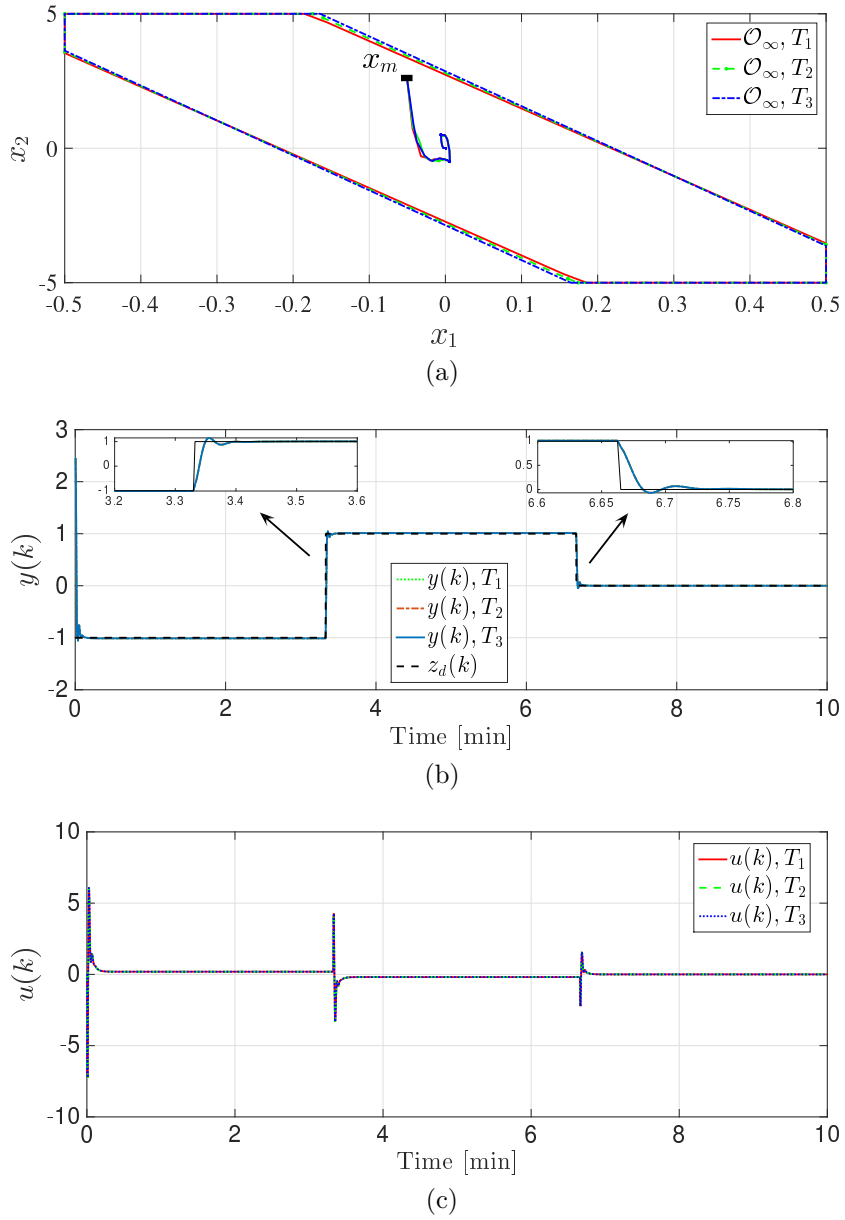


Figure 6.11: (a) Performance of the RDSCTC for different values of the SCT convergence rate T ($T = T_1, T_2, T_3$), to calculate the invariant set $\mathcal{O}_\infty$, on the CSTR plant which was exemplified in [Pannocchia, 2004]; (b) and (c) the plant output and the command signal that resulted from the application of the RDSCTC, respectively.

Table 6.2: The computational time of the RMPC [Pannocchia, 2004].

Control horizon N	1	5	7	11	15	19	25	30
Computational time [sec]	10.55	11.38	17.80	20.22	25.24	31.55	35.25	42.26

above experiment with same configurations was repeated with $T = T_1$ for RDSCTC and with $R = R_1$ for RMPC performance. However, this time different prediction horizons were considered for RMPC. The computational times with the RMPC for different values of N is depicted in Table 6.2. These time values contain both the computation time to find the robustly invariant set and also the time to perform the computations to calculate the control efforts from the initial point until the end of the simulation process. As can be seen in Table 6.2, the total computational time of the RMPC has grown when the control horizon N was increased. However, the simulation time for the RDSCTC in all cases was approximately 15.57 [sec]. Experimentally, it was seen that both RMPC and RDSCTC approximately spend a same computational time to calculate control signal in each sampling interval. This means that, mostly, the differences in computational times in Table 6.2 are due to the computation time the which is needed for the invariant set computation by the two control methodologies.

6.5.2 E. Coli Bioprocess Modeling

Several valuable products such as recombinant proteins, antibiotics and amino acids are nowadays commercially produced through genetically altered microorganisms. In real bioreactors the characteristics of nonlinearity, uncertainty, and disturbance, considered either separately or jointly, can affect the cultivation process and consequently the product maximization. This occurs for instance in Escherichia coli (E. Coli) cultures with aerobic acetate formation. Several studies have been performed for keeping a microbial cultivation process close to its optimized control profile [Hafidi *et al.*, 2008; Dewasmea *et al.*, 2011; Santos *et al.*, 2012]. Traditionally, the control of bioreactors mostly has been limited to pH, temperature, partial pressure of dissolved oxygen, and dissolved carbon dioxide regulations [Diaz *et al.*, 1996]. Without any doubt computer control of the biochemical state variables was an effective way to increase efficiency of the process performance significantly.

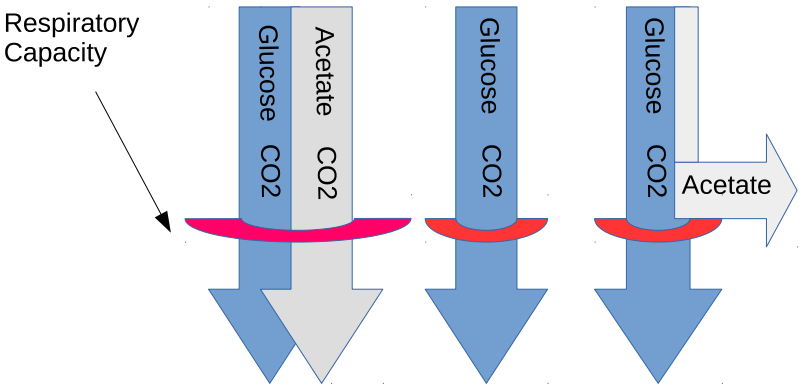


Figure 6.12: Schematic representation of the bottleneck theory.

The growth of the E. Coli bacteria follows the bottleneck theory (Figure 6.12). The E. Coli has a limited respiratory capacity. Acetate is produced when the glucose exceeds the respiratory capacity, which corresponds to the oxido-fermentative regime. Acetate is consumed when glucose is less than the respiratory capacity; which is the oxidative regime. When the quantity of glucose exactly fills the respiratory capacity, the system operates in optimal conditions. This case corresponds to the edge between the two regimes when acetate is not produced nor consumed. It will be seen in further developments that the aim of the control is to force the E. Coli culture to remain at the edge between these two regimes [Rocha, 2003; Hafidi, 2008]. The metabolism of the E. Coli is described through three macroscopic reactions [Hafidi *et al.*, 2008]:

- Glucose oxidation,

$$k_1 S + k_5 O \xrightarrow{r_1} X + k_8 C; \tag{6.64}$$

- Glucose fermentation,

$$k_2 S + k_6 O \xrightarrow{r_2} X + k_9 C + k_3 A; \tag{6.65}$$

- Acetate oxidation,

$$k_4 A + k_7 O \xrightarrow{r_3} X + k_{10} C. \tag{6.66}$$

Table 6.3: Specific growth rates μ_1, μ_2, and μ_3, and expressions of r_X and r_A, depending on the regimes [Hafidi *et al.*, 2008].

Variable	**Oxido-fermentative regime**	**Oxidative regime**
μ_1	$\frac{q_{S,crit}}{k_1}$	$\frac{q_S}{k_1}$
μ_2	$\frac{q_S - q_{S,crit}}{k_2}$	0
μ_3	0	$\frac{q_{AC}}{k_4}$
r_X	$\frac{q_S - q_{S,crit}}{k_2} + \frac{q_{S,crit}}{k_1}$	$\frac{q_S}{k_1} + \frac{q_{AC}}{k_4}$
r_A	$k_3 \frac{q_S - q_{S,crit}}{k_2}$	$-q_{AC}$

where S, O, X, C and A represent glucose (substrate), oxygen, biomass, carbon dioxide, and acetate, respectively. k_i $(i = 1, \ldots, 10)$ are the stoichiometric coefficients. r_j $(j = 1, 2, 3)$ are the growth rates. Reactions (6.64) and (6.66) describe the oxidative regime, and reactions (6.64) and (6.65) describe the oxido-fermentative regime. The relationship between the growth rate r_i and the specific growth rate μ_i $(i = 1, 2, 3)$, which depends on the operation regimes (Table 6.3), is as follows:

$$r_i = \mu_i X. \tag{6.67}$$

Based on the previous scheme, mass balances yield the dynamic model (6.68)-(6.69) [Bastin and Dochain, 1990]:

$$\frac{d}{dt}\begin{bmatrix} X \\ S \\ A \\ O \\ C \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -k_1 & -k_2 & 0 \\ 0 & k_3 & -k_4 \\ 0 & -k_6 & -k_7 \\ k_8 & k_9 & k_{10} \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} - \frac{F_{in}}{W}\begin{bmatrix} X \\ S \\ A \\ O \\ C \end{bmatrix} + \begin{bmatrix} 0 \\ S_{in}\frac{F_{in}}{W} \\ 0 \\ O_{tr} \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ C_{tr} \end{bmatrix}, \tag{6.68}$$

$$\frac{dW}{dt} = F_{in}, \tag{6.69}$$

where W is the culture medium weight [kg], S_{in} is the influent glucose concentration [g/kg], F_{in} is the influent flow rate [kg/h], O_{tr} is the oxygen transfer rate from gas to liquid phase, and C_{tr} the carbon dioxide transfer rate from liquid to gas phase. Considering the bottleneck assumption, then the kinetic terms associated with the glucose consumption q_S, the critical substrate consumption $q_{S,crit}$ (generally dependent on the cells oxidative or respiratory capacity q_O), and the product oxidative rate q_{AC} are expressed by [Santos *et al.*, 2012]:

$$q_S = q_{S,max}\frac{S}{K_S + S}, \tag{6.70}$$

$$q_{S,crit} = \frac{q_O}{k_{OS}} = \frac{q_{O,max}}{K_{OS}}\frac{O}{O + K_O}\frac{K_{i,O}}{K_{i,O} + A}, \tag{6.71}$$

$$q_{AC} = q_{AC,max}\frac{A}{K_A + A}\frac{K_{i,A}}{K_{i,A} + A}, \tag{6.72}$$

where q_O is the respiratory capacity, $q_{S,max}$, $q_{O,max}$, and $q_{AC,max}$ are the maximum specific growth rates, and K_S, K_O, and K_A are the saturation constants of the substrate (glucose), oxygen, and acetate, respectively. K_{OS} is the oxygen yield related to glucose, and $K_{i,O}$ and $K_{i,A}$ are the inhibition constants related to oxygen uptake and acetate uptake, respectively.

Finally, since oxygen is always regulated to induce no influence on the growth of bacteria, the dynamic model (6.68)-(6.69) can be formulated in a compact form as follows [Hafidi *et al.*, 2008]:

$$\frac{dX}{dt} = r_X X - \frac{F_{in}}{W}X, \tag{6.73}$$

$$\frac{dS}{dt} = -q_S X - \frac{F_{in}}{W}\left(S - S_{in}\right), \tag{6.74}$$

$$\frac{dA}{dt} = r_A X - \frac{F_{in}}{W}A, \tag{6.75}$$

$$\frac{dW}{dt} = F_{in}, \tag{6.76}$$

denoted in further developments as $\dot{\mathbf{x}}(t) = f\left(\mathbf{x}(t), F_{in}(t)\right)$, with $\mathbf{x} = \begin{bmatrix} X & S & A & W \end{bmatrix}^T$. r_X and r_A in (6.73)-(6.76) depend on the operating regime and are given in Table 6.3. Further details on the model formulation can be found

in [Hafidi, 2008; Rocha, 2003].

By discretizing (6.73)-(6.76) using the forward differences method with a sampling period of T_s, then the following state-space model (6.77)-(6.78) is obtained for the plant:

$$\begin{aligned}\mathbf{x}(k+1) = \begin{bmatrix} X(k+1) \\ S(k+1) \\ A(k+1) \\ W(k+1) \end{bmatrix} &= \underbrace{\begin{bmatrix} -T_s\frac{X}{W} \\ -T_s\frac{S_{in}-S}{W} \\ -T_s\frac{A}{W} \\ T_s \end{bmatrix}}_{\mathbf{B}(\mathbf{x},k)} F_{in}(k) \\ &+ \underbrace{\begin{bmatrix} 1+T_s r_X & 0 & 0 & 0 \\ -T_s q_S & 1 & 0 & 0 \\ T_s r_A & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}(\mathbf{x},k)} \begin{bmatrix} X(k) \\ S(k) \\ A(k) \\ W(k) \end{bmatrix}, \end{aligned} \tag{6.77}$$

$$\mathbf{y}(k) = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathbf{C}(\mathbf{x},k)} \begin{bmatrix} X(k) \\ S(k) \\ A(k) \\ W(k) \end{bmatrix}, \tag{6.78}$$

where for the recombinant E. Coli cultivation process, the controlled variable will be $\mathbf{y}(k) = y(k) = A(k)$, and the manipulated variable will be $\mathbf{u}(k) = u(k) = F_{in}(k)$.

Maintaining the operating conditions for the optimal biological behavior is attainable if the system works in the edge between the two operating regimes; in this case, the acetate is not produced, leading to $\mu_2 = 0$ and thus $q_S = q_{S,crit}$; and the acetate is not consumed, leading to $\mu_3 = 0$, and thus $q_{AC} = 0$. The optimal feed rate is thus defined for a unique pair of acetate and glucose concentrations, satisfying the aforementioned conditions [Hafidi *et al.*, 2008]:

$$A(k) = 0, \quad S(k) = S_{crit}. \tag{6.79}$$

By applying the conditions of Eq. (6.79) on Eqs. (6.70)-(6.71), then the following relation between the critical substrate concentration level and the cell respiratory

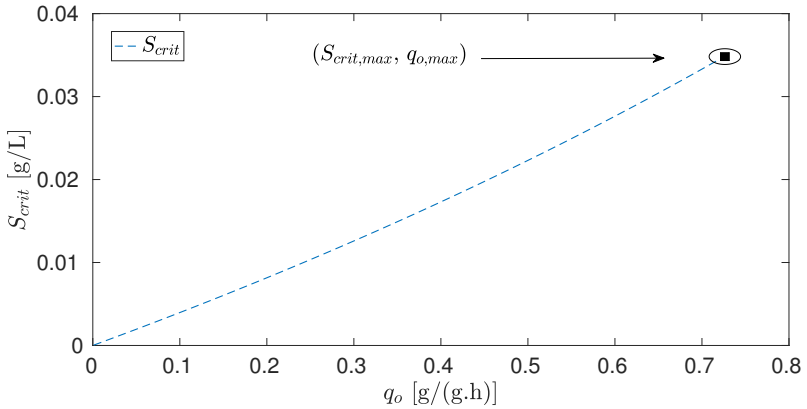


Figure 6.13: S_{crit} as a function of q_O.

capacity is obtained [Santos *et al.*, 2012]:

$$S_{crit} = \frac{K_S q_O}{K_{OS} q_{S,max} - q_O}. \tag{6.80}$$

Figure 6.13 shows how the respiratory capacity q_O has an influence on the critical substrate concentration level, where the point $(0, 0)$ corresponds to a totally inhibited respiratory capacity, preventing any growth, and the point $(q_{O,max}, S_{crit,max})$ corresponds to maximum productivity (i.e., absence of acetate product in the culture medium and a sufficient level of oxygenation). Clearly, the presence of the acetate A in the culture medium can decrease the respiratory capacity, q_O (6.71), and in turn decrease the value of the critical substrate concentration $S = S_{crit}$ (6.80) (Figure 6.13).

In order to maintain the system at the edge between the oxidative (also called respirative) regime and the oxido-fermentative (also called respiro-fermentative) regime, it is necessary to control the acetate, and also keep the substrate concentration S in the culture medium around S_{crit} in order to reach the optimal operating conditions. To this aim, a control strategy achieving a real-time control is needed in order to optimize the process. As can be seen from Eqs. (6.77)-(6.78), the E. Coli plant is modeled in the form of a NSDDTV dynamical system. In the next section, the RDSCTC-PI controller is applied to the NSDDTV E. Coli plant. The design of the RDSCTC-PI for the E. Coli plant is presented assuming that the plant is affected by the presence of a bounded, and unmeasured, additive disturbance.

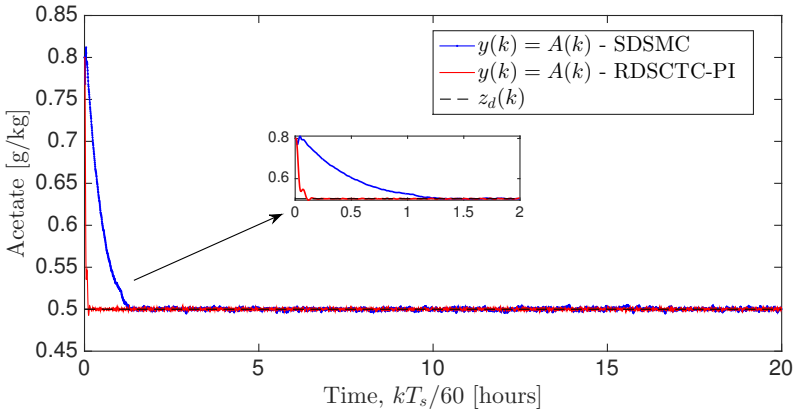
Table 6.4: Parameters values [Hafidi, 2008; Rocha, 2003]

Parameters	**Values**	
k_1	3.164	[g/kg]
k_2	25.220	[g/kg]
k_3	10.900	[g/kg]
k_4	6.382	[g/kg]
$q_{S,max}$	1.832	[g/(kg.h)]
K_S	0.1428	[g/(kg.h)]
K_{OS}	2.020	–
$q_{O,max}$	0.7218	[g/(kg.h)]
$k_{i,O}$	6.952	[g/(kg.h)]
$q_{AC,max}$	0.0967	[g/(kg.h)]
$K_{i,A}$	5.85	[g/(kg.h)]
S_{in}	250	[g/kg]
K_A	0.5236	[g/(kg.h)]
K_{OA}	1.996	[g/(kg.h)]

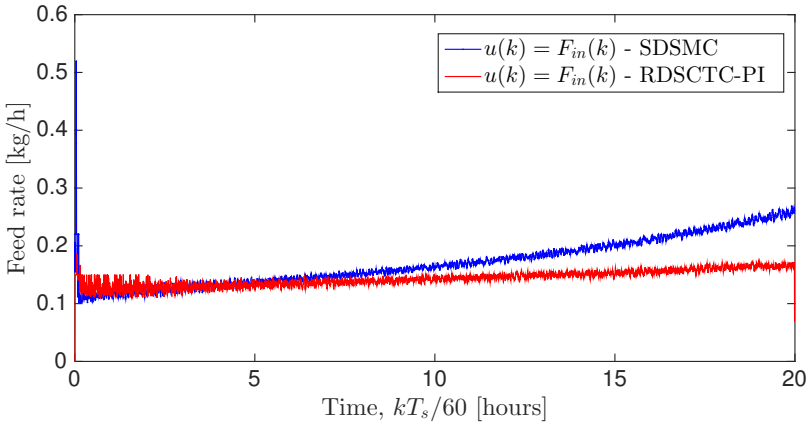
6.5.3 Control of the E. Coli Cultivation Process

For better presentation, the efficiency of the RDSCTC-PI and the SDSMC are tested under two different conditions. In both cases $\mathbf{x}_A = \begin{bmatrix} X_0 & S_0 & A_0 & W_0 \end{bmatrix}^T = [5\,[\text{g/Kg}]; 0.115\,[\text{g/Kg}]; 0.8\,[\text{g/Kg}]; 3.17\,[\text{Kg}]]^T$ is considered as the initial state value. Constant parameters values are considered for all simulations, and are presented in Table 6.4. The sample time is chosen to be $T_s = 2\,[\text{min}]$. The applied $\mathbf{w}(k)$ is a random noise with uniform distribution within the $\mathbf{w}(k)$ bounds defined as $\mathbf{w}(k) = [w_{min} \;\; w_{max}]^T$, where $w_{min} = -[0 \;\; 0 \;\; 0.05 \;\; 0]^T$, and $w_{max} = [0 \;\; 0 \;\; 0.05 \;\; 0]^T$.

Case 1. In a realistic test during a microbial cultivation process, reducing the acetate concentration to an exact zero value is not possible. To make the test condition similar to a real condition, a non-zero A_{ref} was considered for this simulation. The setpoint is $z_d = A_{ref} = 0.5\,[\text{g/Kg}]$ over 20 hours, which is the time required to complete the culture. Figure 6.14 presents the closed loop robust performances of the proposed RDSCTC-PI and of the sliding mode controller (SDSMC). For the RDSCTC-PI scheme $K_P = 1$, $K_I = 0.01$, $T = 1.5\,[\text{min}]$, and $T_s = 2\,[\text{min}]$ were chosen. As the best tuning parameters, for the SDSMC [Eun *et al.*, 1999], $\eta = 0.95$,



(a)



(b)

Figure 6.14: Case 1: (a) Closed-loop simulation results of the output $y(t)$ using the proposed RDSCTC-PI, and standard discrete-time sliding mode controller (SDSMC) [Eun *et al.*, 1999] with the $\mathbf{x}_A$ initial state on the E. Coli plant; and (b) the respective applied command signals.

$q = 0.001$, and $g = 0.8$ were chosen.

As can be seen in Figure 6.14, both controllers can perform the acetate regulation. However, it is clear that, when comparing to SDSMC, a faster response was obtained by the RDSCTC-PI. The best SDSMC result is presented, and was obtained only after the SDSMC was run and tested several times with different SDSMC control parameters sets. Differently, the RDSCTC-PI does not show such a sensitivity in the tuning of the control parameters. When compared to SDSMC the RDSCTC-PI has a much better robust performance in face of changes in the

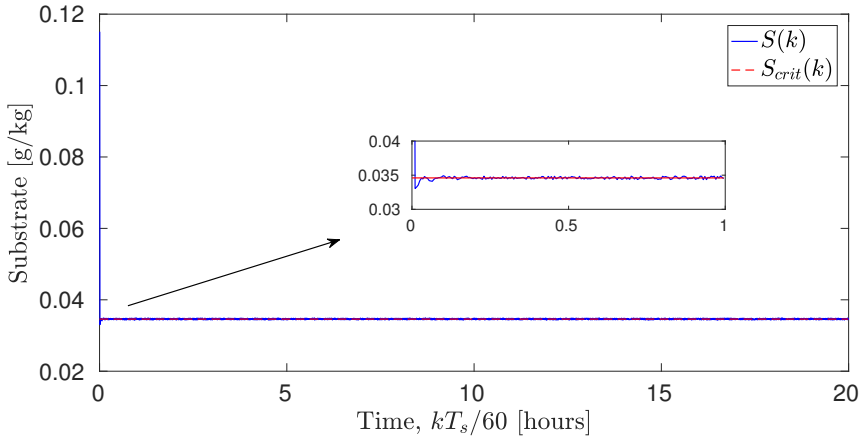


Figure 6.15: Substrate concentration evolution using the RDSCTC-PI controller over 20 hours in the conditions of Case 1 on the E. Coli plant.

controller design parameters. From the performed experiments, and in contrast to the SDSMC sensitivity, it is observed that for the RDSCTC-PI the main control objectives can be attained if condition $0 < T_s/2 < T$ is satisfied.

Figure 6.15 shows the evolution of the substrate state. As can be seen the RDSCTC-PI holds the substrate concentration controlled around S_{crit}.

One of the major drawbacks of other control approaches for the E. Coli process can be the instability of the operating regimes which is generated in the closed-loop control. Such control approaches make the system to permanently switch between different regimes which greatly affects the whole process performances and bacteria production [Hafidi, 2008; Rocha, 2003]. Meanwhile the result presented in Figure 6.16 implies that despite of the presence of unmeasured bounded disturbance the optimal conditions for the operating regime has been maintained by the RDSCTC-PI with full agreement to the control objective for Case 1 ($z_d = A_{ref} = 0.5\,[\text{g/Kg}]$) that corresponds to an oxido-fermentative regime.

From the result in Figure 6.16 it can be deduced that the E. Coli plant operates in the oxido-fermentative regime during all the simulation process (since $\mu_3 = 0$), while it simultaneously keeps the specific growth rate near to zero ($\mu_2 \approx 0$), with respect to a non-zero acetate A, and an unknown disturbance $\mathbf{w}$. As was discussed in Subsection 6.5.2, conditions $\mu_2 = 0$, and $\mu_3 = 0$ are desirable and were considered as main objectives for control purposes in the E. Coli plant, in order to operate in the transition between the two regimes (Table 6.3). The results in Figure 6.16 are

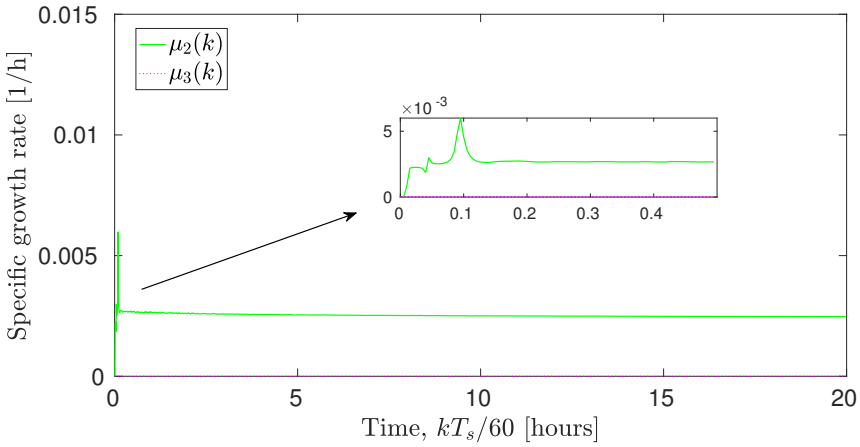


Figure 6.16: Specific growth rate using the RDSCTC-PI controller over 20 hours in the conditions of Case 1 on the E. Coli plant.

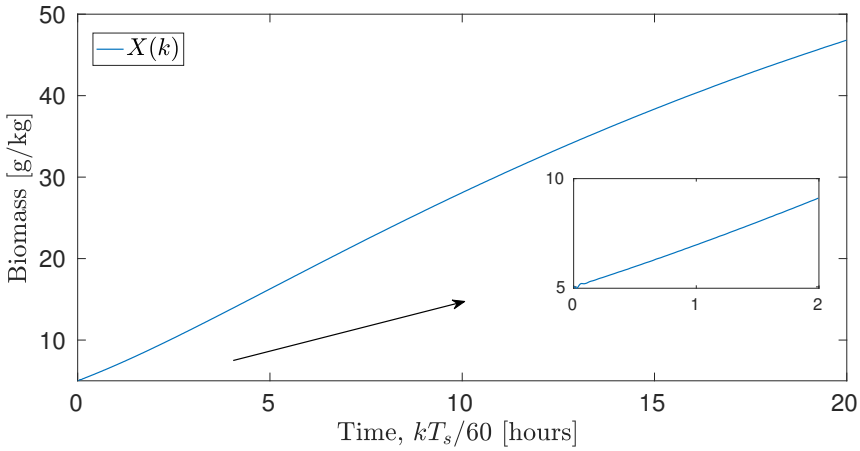


Figure 6.17: Biomass concentration evolution using the RDSCTC-PI controller over 20 hours in the conditions of Case 1 on the E. Coli plant.

close to satisfy the conditions, because $\mu_2 \approx 0$, and $\mu_3 = 0$. The results in Figure 6.16 show that, despite of the presence of unmeasured bounded disturbance, the operating regime has been maintained by the RDSCTC-PI approximately in the border between the two regimes.

Figure 6.17 shows the evolution of the biomass state, where it is shown that the proposed RDSCTC-PI can increase the biomass state, while the plant is exposed to an unmeasured unknown disturbance. Comparing to [Hafidi *et al.*, 2008, Figure 3],

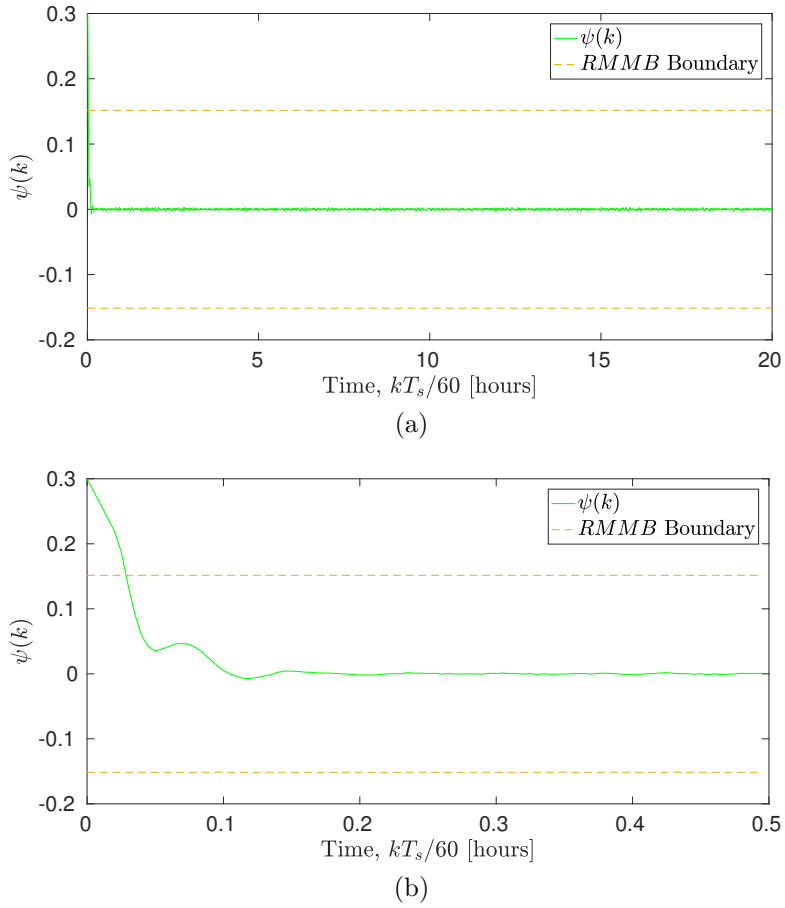


Figure 6.18: Application of the RDSCTC-PI on the E. Coli plant, Case 1: (a) Illustration of the $\psi(k)$ function with the initial state of $\mathbf{x}_A$; and (b) Magnified plot of Figure 6.18(a) for the time period between 0 and 0.5 [hours].

with a same condition, a larger maximum value of the biomass variable is achieved by the RDSCTC-PI. The issue of biomass maximization was handled, while unlike the MPC based controller proposed in [Hafidi *et al.*, 2008], the RDSCTC-PI must also overcome an undesired effect of additive disturbance. The $\boldsymbol{\psi}(k)$ function obtained in the simulation is depicted in Figure 6.18.

A value of $M_\psi = 0.1515$ was obtained from Eq. (6.25). As can be seen in Figure 6.18, after the trajectory has entered the *RMMB*, then with every successive time step, the trajectory remains confined within the *RMMB*.

Case 2. The setpoint is $z_d = A_{ref} = 0.5\,[\text{g/Kg}]$, for $0 < t \leqslant 10\,[\text{hours}]$, while for $10\,[\text{hours}] < t \leqslant 20\,[\text{hours}]$, the reference value changes to $z_d = A_{ref} = 1\,[\text{g/Kg}]$. Also, unexpected plant parameter changes are applied: two constant parameters $S_{in} = 250\,[\text{g/kg}]$ and $k_{i,o} = 6.952\,[\text{g/kg}]$ are changed to $S_{in} = 500\,[\text{g/kg}]$, and $k_{i,o} = 3.476\,[\text{g/kg}]$, respectively, for $5\,[\text{hours}] \leqslant t \leqslant 15\,[\text{hours}]$. In the scenario of this Case 2, $K_P = 0.58$, $K_I = 0.02$, $T = 1.06\,[\text{min}]$, and $T_s = 2\,[\text{min}]$ were chosen in the RDSCTC-PI. For the SDSMC [Eun *et al.*, 1999], $\eta = 0.8$, $q = 0.005$, and $g = 0.6$ were chosen as the best tunning parameters. Note that the change of acetate setpoints for this E. Coli plant example is not of practical relevance, and the goal of the regulation is to track a value very close to zero. In fact, this Case 2 scenario shows the adequate tracking behavior of the developed RDSCTC-PI control strategy.

Figure 6.19 shows the impact of a change of acetate concentration setpoints in this Case 2. The results of Figure 6.19(a) confirm that comparing to the SDSMC, a faster response was obtained by the RDSCTC-PI, including a faster response of RDSCTC-PI to the change of setpoint at time $t = 10\,[\text{hours}]$. Figure 6.20 shows the evolution of the substrate state, where it is confirmed, in the new conditions of Case 2, the good tracking and robustness performances of the RDSCTC-PI. In fact, the substrate concentration S can track S_{crit} well, even with unexpected changes in both the plant output setpoints (tracking) and on the plant model parameters (robustness). Furthermore, during the interval $5\,[\text{hours}] < t \leqslant 15\,[\text{hours}]$ where the plant parameters were changed/different, it can be seen that unlike RDSCTC-PI, the SDSMC performance imposes a large value of command law variation (see Figure 6.19(b)). Clearly, in the case that the E. Coli plant faces constraints on Δu, then the large variation of the required actuation commands can limit the SDSMC performance.

Figure 6.21 shows the evolution of the biomass state for the RDSCTC-PI case. The biomass value at the end of the simulation at time $t = 20\,[\text{hours}]$ in Figure 6.21, when compared with the result in Figure 6.17, confirms that, as expected, increasing acetate A further away from zero can reduce the amount of maximum biomass state X that is attained at the end of the E. Coli cultivation process experiment. Also, the increase of acetate concentration, A, reduced the level of S_{crit} (see Figure 6.20). The $\boldsymbol{\psi}(t)$ function obtained in the simulation of Case 2 is depicted in Figure 6.22. A value of $M_\psi = 0.530$ was obtained from Eq. (6.25).

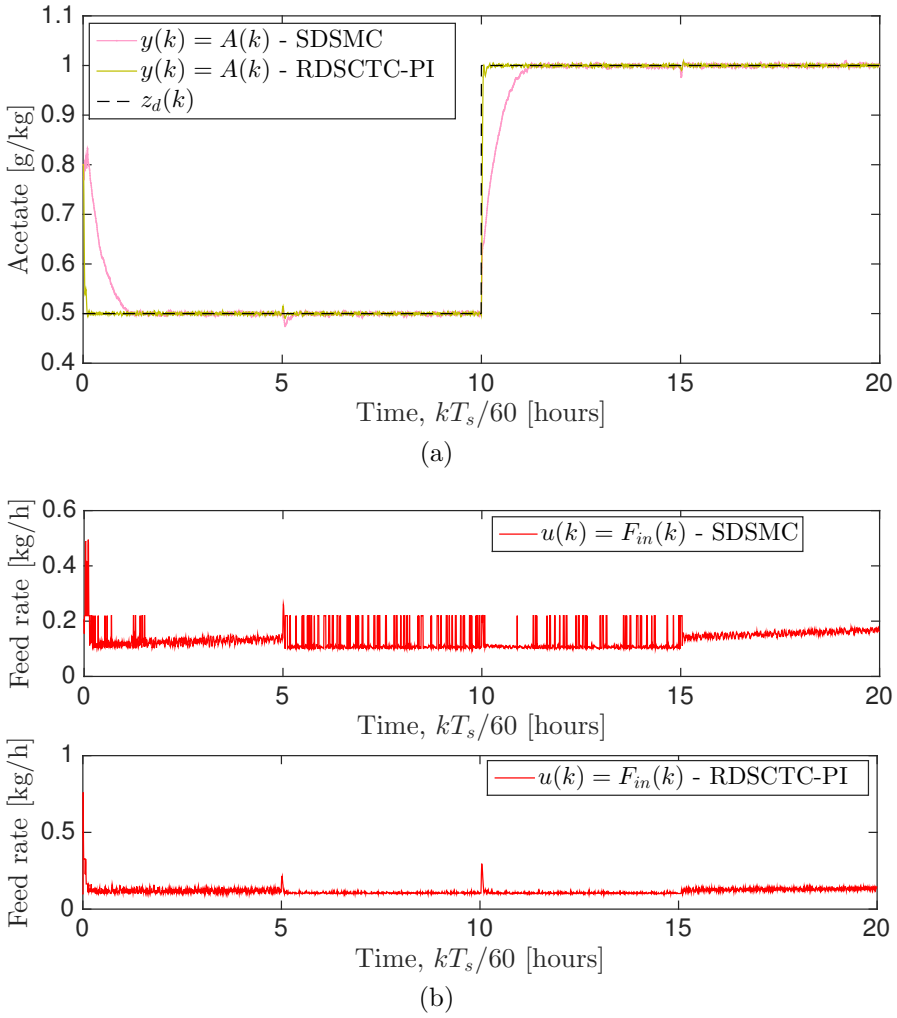


Figure 6.19: Application of the RDSCTC-PI and the SDSMC controllers on the E. Coli plant, Case 2: (a) Closed-loop simulation results of the output $y(t)$, for the $\mathbf{x}_A$ initial state; and (b) the respective command signals applied by the two controllers.

In another test for Case 2, it was tried to achieve a faster response to setpoint changes by the SDSMC. A faster response was obtained after several trial-and-error runs of the SDSMC algorithm. However, the chattering problem has emerged for the attained faster response case: Figure 6.23 shows how the output in the fed-batch bioprocesses can be affected by the chattering phenomenon. Moreover, the result of the SDSMC performance was obtained without including the additive disturbance. In other words, the frequent variations in the output in this SDSMC experiment are

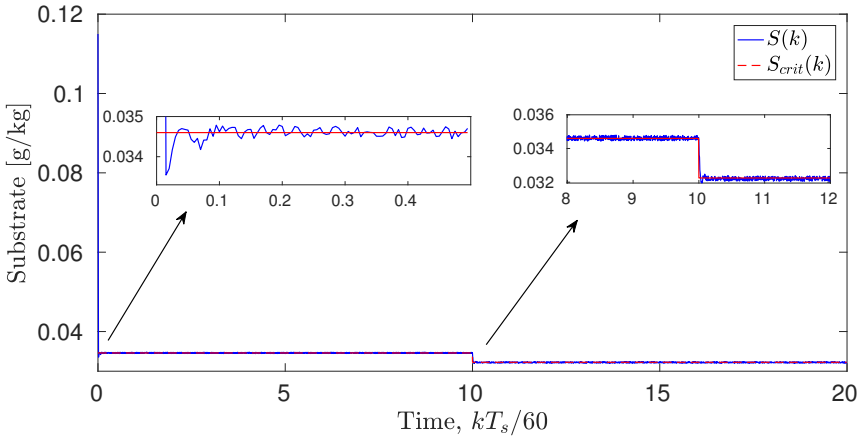


Figure 6.20: Substrate concentration evolution using the RDSCTC-PI controller over 20 hours in the conditions of Case 2 on the E. Coli plant.

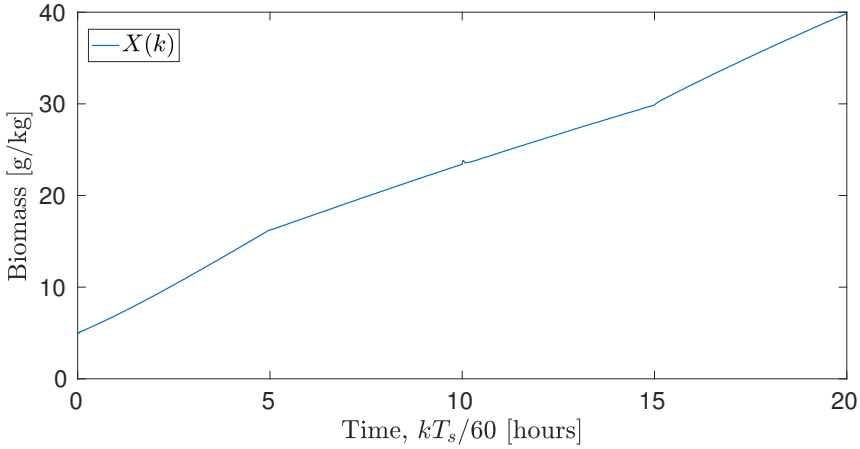


Figure 6.21: Biomass concentration evolution using the RDSCTC-PI controller over 20 hours in the conditions of Case 2 on the E. Coli plant.

not caused by disturbances, but rather they are caused by the switching tendency of the SDSMC control law.

6.5.4 Control of Two Coupled DC Motors

In this section, both SCT based controllers proposed in this Chapter, i.e. RDSCTC, and RDSCTC-PI as well as the two adaptive fuzzy controllers proposed in Chapter. 5, i.e. AFGPC (Method 1), and AFGPC (Method 2), are tested on the two coupled

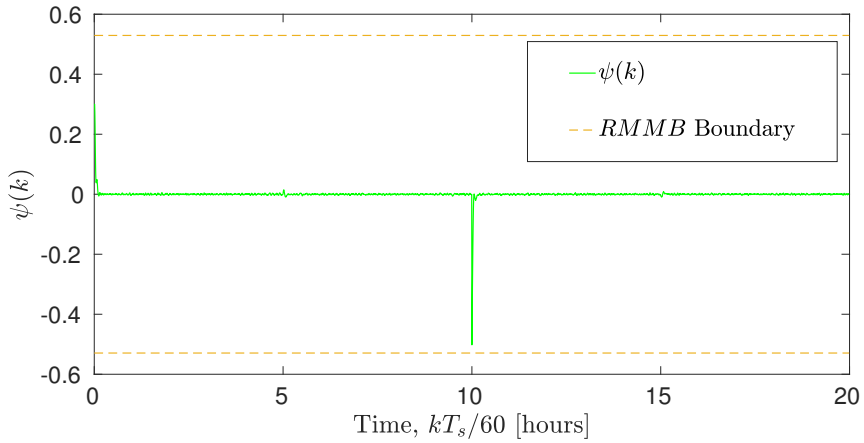


Figure 6.22: Application of the RDSCTC-PI on the E. Coli plant, Case 2: Illustration of the $\boldsymbol{\psi}(t)$ function for the $\mathbf{x}_A$ initial state for the reference signal.

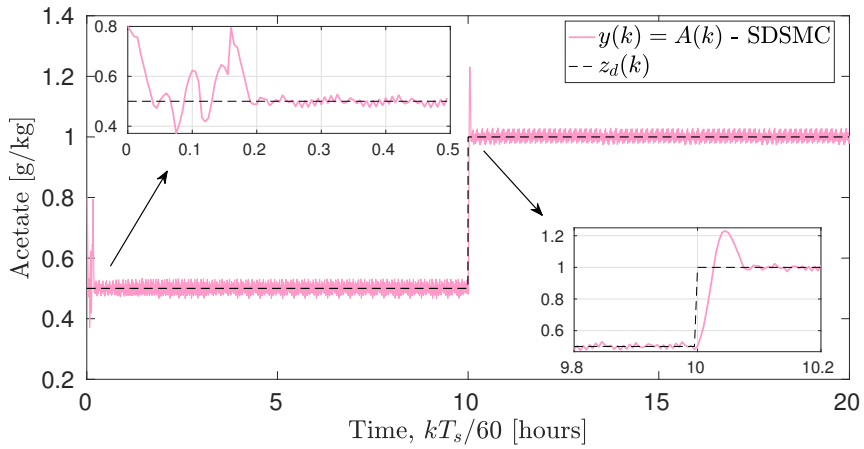


Figure 6.23: Chattering phenomena incurred for obtaining a faster SDSMC response on the fed-batch E. Coli bioprocess.

DC motors experimental setup (see Subsection 4.4.4).

A simulation stage was implemented in order to analyze different operating conditions and to compare with real-time experiments. Using electromagnetic and mechanical laws, and considering the case where the load disturbance applied to the generator (motor 2) is $R_L < \infty$, the following continuous-time state space model of

the plant is obtained:

$$\begin{bmatrix} i_1(k+1) \\ i_2(k+1) \\ \omega(k+1) \end{bmatrix} = \begin{bmatrix} 1-\frac{T_s R_{t1}}{L_{t1}} & 0 & -\frac{T_s K_{v1}}{L_{t1}} \\ 0 & 1-\frac{T_s(R_{t2}+R_L)}{L_{t2}} & \frac{T_s K_{v2}}{L_{t2}} \\ \frac{T_s K_{t1}}{J_{t1}+J_{t2}} & \frac{T_s K_{t2}}{J_{t1}+J_{t2}} & 1-\frac{T_s(B_{t1}+B_{t2})}{J_{t1}+J_{t2}} \end{bmatrix} \begin{bmatrix} i_1(k) \\ i_2(k) \\ \omega(k) \end{bmatrix} + \begin{bmatrix} \frac{T_s}{L_{t1}} \\ 0 \\ T_s\zeta_t \end{bmatrix} V_1(k) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \vartheta(k), \tag{6.81}$$

$$\omega(k) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i_1(k) & i_2(k) & \omega(k) \end{bmatrix}^T. \tag{6.82}$$

where V_1 is the voltage source for motor 1, ω is the angular velocity of motors 1 and 2, and i_i, R_{ti}, L_{ti}, B_{ti}, J_{ti}, K_{vi}, and K_{ti} are the armature current, armature resistance, inductance, viscous friction coefficient, moment of inertia, velocity constant (or back electromotive (EMF) constant), and torque constant of motor i ($i = 1, 2$), respectively. ϑ and $\zeta(t)$ are external and internal disturbances, respectively. The sampling period in the experiment with the simulated physical model was the same as the one used in the real-world experiment, $T_s = 0.1\,[\text{s}]$.

The model parameters were identified from data obtained from the real motor (Figure 4.10(a), Section 4.4.4), and using an optimization algorithm to search for the best fit of the parameters. The motor parameters obtained were $L_{ti} = 1.95074\,[\text{H}]$, $R_{ti} = 1.4020\,[\Omega]$, $J_{ti} = 8 \times 10^{-6}\,[\text{Kg}{\cdot}\text{m}^2]$, $B_{ti} = 6.3445 \times 10^{-6}\,[\text{N}{\cdot}\text{m}{\cdot}\text{s/rad}]$, $K_{vi} = 0.0040\,[\text{V/rad/s}]$, $K_{ti} = 0.0048\,[\text{N}{\cdot}\text{m/A}]$, for $i = 1, 2$. Experimentally, in the real test it was seen that the output error is greater when the input signal increases, due to the fact that the shaft which connects the two motors is wraped and twisted, which produces nonlinearities and/or a time varying load with non-uniform characteristics for different motor angles. Thus, an interconnection disturbance namely, $\zeta_t = 0.1$, was considered in (6.81) to model this situation.

By discretizing model (6.81)-(6.82) using the forward differences method, and using the obtained motor parameters on both the two coupled DC motors, model

(6.81)-(6.82) is rewritten as

$$\mathbf{x}(k+1) = \begin{bmatrix} 0.9281 & 0 & -0.0021 \\ 0 & 0.9281 - (R_L/19.5074) & 0.0021 \\ 30.0000 & 30.0000 & 0.9207 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 0.0512 \\ 0 \\ 0.01 \end{bmatrix} \mathbf{u}(k) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \vartheta(k), \tag{6.83}$$

$$y(k) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{x}(k), \tag{6.84}$$

where $\mathbf{x}(k) = [x_1(k), x_2(k), x_3(k)]^T = [i_1(k), i_2(k), \omega(k)]^T$, and $\mathbf{u}(k) = V_1(k)$. The following admissible disturbance is considered: $-2 \leqslant \vartheta(k) \leqslant 2$. The objective of the experiment is to control the angular velocity asymptotically by manipulating of the voltage source for motor 1.

For the RDSCTC-PI control scheme $K_P = 1.0$, $K_I = 0.095$, $T = 0.06\,[\text{s}]$, and $T_s = 0.1\,[\text{s}]$ were chosen. The same values of the T, and T_s parameters were considered for RDSCTC, i.e. $T = 0.06\,[\text{s}]$, and $T_s = 0.1\,[\text{s}]$. For the AFGPC (Method 1), the following controller parameters were chosen: $N_p = 25$, $N_u = 1$, $\lambda = 8$, $\bar{d} = 0$, $\rho = 0.95$, $\varphi_i = 1$, $\tau_i = 1 \times 10^{-3}, \nu_i = 1 \times 10^{-6}$, for $i = 1, \ldots, c$, and $c = 9$. For the AFGPC (Method 2), the following controller parameters were chosen by the user: $N_p = 25$, $N_u = 1$, $\lambda = 8$, $\bar{d} = 0$, $w_{r,max} = 0.9$, $w_{r,min} = 0.4$, $c_{1r} = 1.5$, $c_{2r} = 2$, $r_{max} = 20$, $v_{max} = 1.0$, $v_{min} = -0.8$, $\gamma = 1$, $\beta = 0.1$, and $T_2 = 20$. The best control parameters for both AFGPC methodologies were found experimentally by trial and error, and considering the results obtained in Subsection 5.3.2.

The setpoint of the controlled variable, $z_d(t)$, is changed as follows during the simulations:

$$z_d(k) = \begin{cases} 350, & 0 < k \leqslant 150\,, \\ 400, & 150 < k \leqslant 400\,, \\ 380, & 400 < k \leqslant 600\,, \end{cases} \tag{6.85}$$

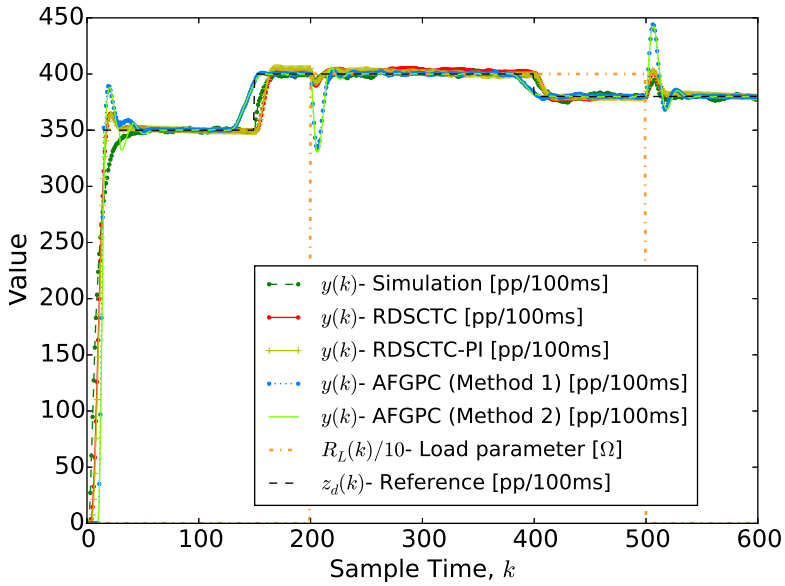
Table 6.5: Numerical results of the real and simulation tests obtained by the proposed RDSCTC, RDSCTC-PI, AFGPC (Method 1), and AFGPC (Method 2) for the two coupled DC motors setup. Note that, here, the $MSE = \frac{1}{T_n}\sum_{k=1}^{T_n}(y(k) - z_d(k))^2$ is related to the tracking error. T_n is the total number of samples on the test. The values were multiplied by 10^4.

Methodology	$1/MSE$
Simulation	1.4783
RDSCTC	1.2603
RDSCTC-PI	1.5967
AFGPC (Method 1)	1.3033
AFGPC (Method 2)	1.3698

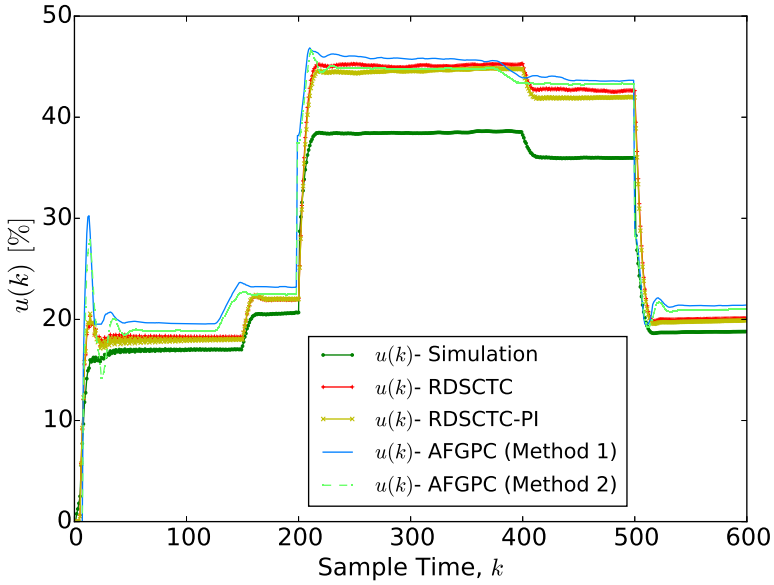
and the load disturbance $R_L(k)$ is

$$R_L(k) = \begin{cases} 0, & 0 < k \leqslant 200\,, \\ 4000, & 200 < k \leqslant 500\,, \\ 0, & 500 < k \leqslant 600\,. \end{cases} \tag{6.86}$$

Figure 6.24 presents the results of the closed-loop performance on both the simulation and the real test obtained by the proposed RDSCTC, RDSCTC-PI, AFGPC (Method 1), and AFGPC (Method 2) controllers, starting from an initial state of zero. Figure 6.24 contains results from performance of all the four aforementioned controllers on the real two coupled DC motors setup. For the simulated model of the two coupled DC motors setup, in order to have an illustrative figure and avoid confusion, among all results of the four controllers, just the result obtained by the RDSCTC was depicted in Figure 6.24. In Figure 6.24(a), the value of $R_L(k)$ is presented after multiplying it by a factor of 0.1. Table 6.5 shows numerical results of the four real and RDSCTC simulation tests obtained by the aforementioned control methodologies for the two coupled DC motors setup. The results in Table 6.5 show that the RDSCTC-PI had a better performance to track setpoints. When the load disturbance is applied at $200 \leqslant k \leqslant 500$, there is an undershoot at $k = 200$ and an overshoot at $k = 500$ in the system response. As can be seen in Figure 6.24(a), larger undershoots at $k = 200$ and larger overshoots at $k = 500$ have been obtained by the AFGPC controllers when the results are compared to SCT based controllers. At the



(a)



(b)

Figure 6.24: (a) Closed loop real and simulation results obtained by the proposed RDSCTC, RDSCTC-PI, AFGPC (Method 1), and AFGPC (Method 2) for the two coupled DC motors setup, starting from an initial state of zero; and (b) the respective applied command signals.

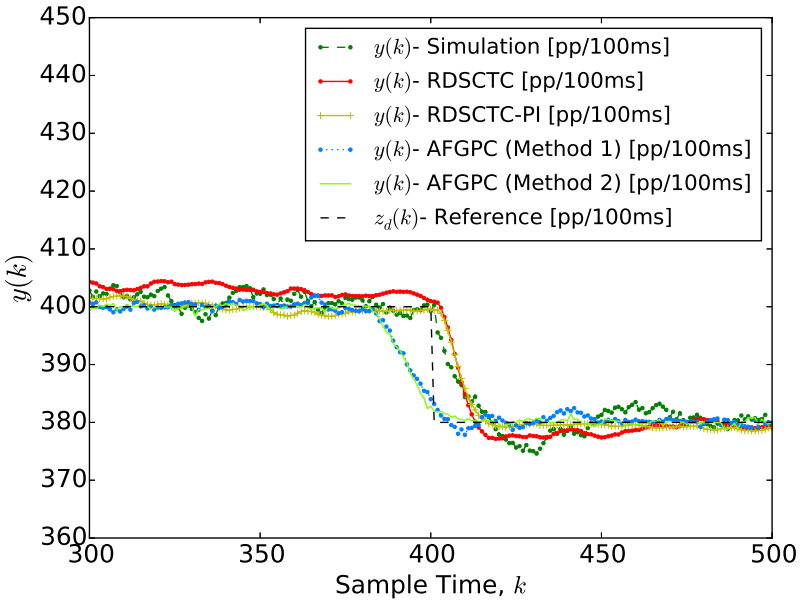


Figure 6.25: Magnified plot of Figure (a) for the time period between 300 and 500.

beginning, at $k = 20$ an overshoot occurred when using the AFGPC methodologies. Experimentally, it was realized that larger values of λ can reduce the overshoot value at $k = 20$. However, after that it was seen that a larger value of λ will increase the values of undershoot, and overshoot at $k = 200$, and at $k = 500$, respectively. Figure 6.25 presents a magnified plot of Figure 6.24(a) for the time period between 300 and 500. The results in Figures 6.24 and 6.25 confirm that AFGPC controllers had an acceptable performance to track $z_d(k)$ in the DC motors setup. The exception is at the initial step after $k = 0$, and at $k = 200$ and $k = 500$, where larger undershoots and overshoots can be seen in the results of the AFGPC controllers, otherwise in most of the times, the output tracking results obtained by the AFGPC controllers are comparable to the tracking results obtained by the SCT-based controllers.

6.6 Conclusion

In this chapter, first the problems/concerns regarding the control of DT-LPV systems were investigated. An innovative robust constrained control framework for DT-LPV systems, was proposed based on the synergetic control theory (SCT) technique and the invariant set theory. The proposed RDSCTC controller was used

to drive the state from any initial state inside the robustly positively invariant set to desired and allowable setpoints. It was shown that in DT-LPV systems without model uncertainty, and for any unmeasured bounded additive disturbance, the proposed controller stabilizes the system by asymptotically driving the error of the controlled variable to a bounded set containing the origin and then maintaining it there. Moreover, given an uncertain DT-LPV system jointly subject to unmeasured and constrained additive disturbances, and constraints in states, input commands, and reference signals (setpoints), then invariant set theory was used to derive an appropriate polyhedral robust invariant region in which the proposed control framework was guaranteed to robustly stabilize the closed-loop system. Furthermore, the proposed controller can track varying non-zero control setpoints even in the presence of complex conditions such as disturbances, fast setpoints changes, and the uncertainty in the DT-LPV plant. The controller is characterized to have a simple structure leading to an easy implementation, and a non-complex design process. The robust performance and effectiveness of the proposed controller and the implications of the controller design on feasibility and closed-loop performance were demonstrated through application examples on the temperature control of a continuous stirred tank reactor (CSTR) plant. To better demonstrate the efficiency of the RDSCTC, the closed-loop simulation results obtained by the proposed RDSCTC were demonstrated and compared with the performance of the RMPC [Pannocchia, 2004]. Three points were noticeable when the RDSCTC was compared with the RMPC. First, a larger invariant set region was ensured/obtained by the RDSCTC. Second, the RDSCTC showed less sensitivity to unexpected control parameter changes when compared to the RMPC. Third, for prediction horizons larger than $N = 0$ the computational time required in the RMPC for the invariant set computation is experimentally observed to grow in order to have an enlarged invariant set region. Moreover, the RDSCTC did not show such sensitivity of the computational time as a function of N.

In the continuation of this Chapter, attention was paid to the issue of nonlinear state dependent discrete time varying (NSDDTV) systems control. A controller was proposed for NSDDTV systems in the presence of unknown additive disturbances. The proposed discrete-time SCT control framework uses a proportional-integral (PI) type of the SCT macro-variable and a one-step delayed estimation of the disturbance. The proposed controller was characterized to have robust properties and advantages

over SMC, but, unlike SMC, without suffering from a chattering phenomenon. As the case study, the controller was applied to the fed-batch fermentation of E. Coli system. The objective was to use the proposed controller to control the acetate concentration on the E. Coli cultivation bioprocess in order to maximize the biomass concentration at the end of the process by adjusting the amount of the feed given to the system. The proposed controller can track acetate setpoints in the presence of complex conditions such as disturbances, fast setpoints changes, and system parameters variations. Results revealed that comparing to the SMC controller, the proposed method performed a faster response to setpoints changes.

Finally, to gather results from all controllers proposed in this thesis, both the SCT based controllers, i.e. RDSCTC, and the controller based on a PI-type SCT macro-variable which is called RDSCTC-PI, as well as both the two adaptive fuzzy model predictive controllers proposed in Chapter 5, i.e. AFGPC (Method 1), and AFGPC (Method 2), were tested on the two coupled DC motors experimental setup and control results were discussed.

Chapter 7

Conclusions

Nowadays a vast amount of changes can be seen in industrial processes comparing with past decades. There are enormous economic incentives to optimize industrial processes. World's attraction from traditional old factory layouts to modern industrial designs increased concerns regarding environmental pollution, and global energy consumption. Both aforementioned concerns have direct connection with natural resources degradation. In industry, lack of high accuracy on system parameters measuring in instrumentation could intensify these global problems. Large and significant attention to these concerns increased a vast worldwide competition and research, not only in trying to fabricate more standard devices and instruments, but also in control techniques applicable to achieve optimal control designs for industrial processes. However, the design of an optimal controller which simultaneously can satisfy control purposes on a real process, is not an easy task. The presence of complex characteristics in industrial processes such as nonlinearities, unknown and time-varying dynamics, constraints, disturbances, and uncertainties increases the difficulty in control designs.

Mostly, a good control design for a nonlinear plant can be obtained if for that plant a good model can be estimated. Several types of approaches to modeling nonlinear plants can be considered to be used in control design. Among them, fuzzy modeling has become an active research area because of the capability of handling perceptual uncertainties, such as vagueness and ambiguity, involved in the interpretation of a real system. Also, it has shown excellent ability when describing nonlinear systems, in particular with the Takagi-Sugeno (T-S) fuzzy models. The

Takagi-Sugeno fuzzy models with simplified linear rule consequents are universal approximators capable of approximating any continuous nonlinear system. However, the T-S fuzzy models have important components which need to be computed well. Some important components for a T-S fuzzy model structure are the consequent parameters, antecedent parameters, membership functions types, as well as the fuzzy rules and fuzzy sets. In this context, the use of computational intelligence techniques can be useful to find the T-S fuzzy model parameters. Also, the presence of an external disturbance on the system as well as other concerns such as uncertainties, and system constraints can impose restrictions in the control design which must be considered. In this situation, a new class of control design to handle systems subject to the presence of such concerns must be considered.

Motivated by these problems, three main research directions were addressed in this thesis: automatic identification of T-S fuzzy models, design of frameworks for model predictive control purposes without having any prior knowledge about a model of the plant, and robust control designs for uncertain systems subject to system constrains, and external disturbances.

In Chapter 4, the design of two different novel online evolving Takagi-Sugeno (T-S) fuzzy model identification methodologies was considered. The first proposed T-S identification methodology used a new unsupervised fuzzy clustering algorithm (NUFCA), and the RLS-ADF method to construct an online evolving T-S fuzzy model identification method. The proposed system identification approach consists of two main steps: antecedent T-S fuzzy model parameters identification and consequent parameters identification. The NUFCA combines the K-nearest neighbor and fuzzy C-means methods into a fuzzy modelling method for partitioning the input data and identifying the antecedent parameters of the fuzzy system; then the RLS-ADF method is exploited to obtain initial consequent parameters and to construct a method for on-line T-S fuzzy model identification.

As second part of the work related to the T-S fuzzy identification methodologies, a hierarchical particle swarm optimization (HPSO) algorithm was introduced to automatically extract all fuzzy logic system (FLS)'s parameters of a T-S fuzzy model. The HPSO algorithm is composed of a six level structure, where the first level is responsible for the selection of an adequate set of input variables and delays. The second level considers the antecedent membership functions. The consequent parameters are defined on the third Level. Level four is devoted to particles of

individual rules. The set of rules are obtained on the fifth level, and finally, the sixth level is appointed as the control unit of the FLS, specifying the choice of several configurations in the lower levels. For improving the convergence time, the NUFCA was employed to initialize particles positions inside the HPSO, as well as the number of partitions. Finally, an online adaptive approach based on the HPSO was proposed which results in a self-adaptive HPSO (S-AHPSO) approach to make an online learning/updating of the fuzzy T-S model.

To validate and demonstrate the performance and effectiveness of the two proposed identification methodologies, the identification of the dissolved oxygen in an activated sludge reactor within a simulated wastewater treatment plant (WWTP), the identification of a real WWTP plant, and the identification of a real-world experimental setup composed of two coupled DC motors were studied. The results have shown that the proposed techniques in both methodologies can successfully identify fuzzy model parameters that represent the dynamics of nonlinear plants, using only a data set of the process, where the model can be further used to estimate the output of the plant. Moreover, the results revealed superior performance of the proposed methods when compared to other state of the art methods.

In Chapter 5, the design of new methodologies for fuzzy model predictive control of nonlinear time-varying systems without the knowledge about the mathematical model of the plant, were considered. The T-S fuzzy system learned by the T-S fuzzy model identification algorithms proposed in Chapter 4 was integrated into the control domain to construct an effective adaptive fuzzy generalized predictive controller (AFGPC).

To validate and demonstrate the performance and effectiveness of the proposed adaptive model predictive controllers, they were tested in the problems of controlling the dissolved oxygen in the activated sludge reactor within a simulated wastewater treatment plant (WWTP); and on the control of the real-world experimental setup composed of two coupled DC motors. Results have shown that the proposed controller methodologies can control processes using only a data set of the process to initialize the adaptive T-S fuzzy models. The experiments revealed that the total proposed AFGPC adaptive controller frameworks outperform typical well-known control methods such as the PID controller, and the classical GPC controller.

In Chapter 6, the RDSCTC, a novel robust constrained control methodology for discrete-time linear parameter varying (DT-LPV) systems, was proposed based on

a synergetic control theory (SCT) approach and on the invariant set theory. The proposed controller, can be used to drive the state from any initial state inside the robustly positively invariant set to desired and allowable setpoints. Stability analysis for the RDSCTC was discussed and a theorem was given. It was shown that in DT-LPV systems without model uncertainty, and for any unmeasured bounded additive disturbance, the proposed controller stabilizes the system by asymptotically driving the error of the controlled variable to a bounded set containing the origin and then maintaining it there. Moreover, given an uncertain DT-LPV system jointly subject to unmeasured and constrained additive disturbances, and constraints in states, input commands, and reference signals (setpoints), then invariant set theory was used to derive an appropriate polyhedral robust invariant region in which the proposed control framework was guaranteed to robustly stabilize the closed-loop system. Furthermore, the proposed controller can track varying non-zero control setpoints even in the presence of complex conditions such as disturbances, fast setpoints changes, and the uncertainty in the DT-LPV plant. The controller is characterized to have a simple structure for implementation, and a non-complex design process. The robust performance and effectiveness of the proposed controller and the implications of the controller design on feasibility and closed-loop performance were demonstrated through application examples on the temperature control of a continuous stirred tank reactor (CSTR) plant.

In the continuation of Chapter 6, attention was paid to the issue of NSDDTV systems control. A controller for NSDDTV systems in the presence of unknown additive disturbances, was presented. The proposed discrete-time SCT control framework uses a proportional-integral (PI) type of the SCT macro-variable and a one-step delayed estimation of the disturbance. The proposed controller was characterized to have robust properties and advantages of SMC, but without inheriting the SMC chattering phenomena. As the case study, the controller was applied to the fed-batch fermentation of the E. Coli system. The objective was to use the proposed controller to control the acetate concentration on the E. Coli cultivation bioprocess in order to maximize the biomass concentration at the end of the process by adjusting the amount of the feed given to the system. The proposed controller could track acetate setpoints in the presence of complex conditions such as disturbances, fast setpoints changes, and system parameters variations. Results showed that comparing with the SMC controller, the proposed method presented a faster response to setpoints

changes.

7.1 Future Work

As future works the following issues can be investigated:

- A more effective unsupervised algorithm can be investigated to find the important T-S fuzzy model components such as the best membership functions types, the best input variables, the fuzzy rules, as well as the parameters of the antecedent fuzzy membership functions.

- Furthermore, a new adaptive procedure for online identification can be investigated to be used for an adaptive online model predictive control design. Considering the fact that an online system identification plays an important role in online control design, thus, investigation in order to find a more accurate method usable for adaptive process identification is worth further research.

- In this thesis, a hierarchical particle swarm optimization (HPSO) algorithm was introduced to automatically extract the structure and all fuzzy logic system (FLS)'s parameters of a T-S fuzzy model. A new technique based on the hierarchical particle swarm optimization algorithm proposed in this thesis, can also be investigated to be used in direct fuzzy control design.

- Regarding the issue of invariant set design which was discussed in this thesis, a new approach can be investigated to extend the initial invariant set region. Enlarging the set of initial plant states in which the proposed control framework is guaranteed to robustly stabilize the closed-loop system, and perform asymptotical setpoints tracking, is a relevant future research direction. An enlarged invariant set region gives the possibility of a wider range of initial states selection for the designer. This issue is important because mostly, in real world applications, and depending on different practical situations, different initial values are required to initialize a control process.

- An adaptive fuzzy system was studied in this thesis to construct an adaptive online model predictive controller. However, neither the adaptive procedure nor the fuzzy system were integrated with the SCT control design in this thesis.

Therefore, an adaptive fuzzy SCT control approach can be a relevant subject to study for the future work.

- Finally, new controllers based on SCT have been designed and applied on both DT-LPV and NSDDTV systems in this thesis. SCT-based control designs for SISO systems have been considered and stability analyses were given. The development of SCT-based control designs, and corresponding stability analyses, for MIMO systems is a relevant direction for future work.

Bibliography

[Abbasi *et al.*, 2015] R. Abbasi, M. T. Hamidi Beheshti, and M. Mohraz. Modeling and Stability Analysis of HIV-1 as a Time Delay Fuzzy T-S System via LMIs. *Applied Mathematical Modelling*, vol. 39, no. 23, pp. 7134–7154, 2015. (Cited in page 40).

[Abdelmalek *et al.*, 2016] Samir Abdelmalek, Linda Barazane, Abdelkader Larabi, and Maamar Bettayeb. A Novel Scheme For Current Sensor Faults Diagnosis in the Stator of a DFIG Described By a T-S Fuzzy Model. *Measurement*, vol. 91, pp. 680–691, September 2016. (Cited in page 40).

[Al-Othman and Irving, 2007] A. K. Al-Othman and M. R. Irving. Robust State Estimator Based on Maximum Constraints Satisfaction of Uncertain Measurements. *Measurement*, vol. 40, no. 3, pp. 347–359, April 2007. (Cited in page 99).

[Alfi and Modares, 2011] Alireza Alfi and Hamidreza Modares. System Identification and Control Using Adaptive Particle Swarm Optimization. *Applied Mathematical Modelling*, vol. 35, no. 3, pp. 1210–1221, 2011. (Cited in pages 39, 60, 62, 63, 68, 71, 73, and 75).

[Ali and Ramaswamy, 2009] Sk. Faruque Ali and Ananth Ramaswamy. Optimal Fuzzy Logic Control for MDOF Structural Systems Using Evolutionary Algorithms. *Engineering Applications of Artificial Intelligence*, vol. 22, no. 3, pp. 407–419, April 2009. (Cited in page 39).

[Antsaklis and Michel, 2007] Panos J. Antsaklis and Anthony N. Michel. *A Linear Systems Primer*. Birkhäuser, Boston, MA, USA, 2007. (Cited in pages 106, and 121).

[Åström and Wittenmark, 1997] Karl J. Åström and Björn Wittenmark. *Computer-Controlled Systems: Theory and Design*. Prentice–Hall, Inc, Upper Saddle River, NJ, USA, 1997. (Cited in page 122).

[Bai *et al.*, 2012] Quanyong Bai, Dongxiao Niu, and Jia Bai. Complex Networks Economy Systems Engineering in General Synergetic Structure. *Systems Engineering Procedia*, vol. 4, pp. 252–258, 2012. (Cited in page 100).

[Barros and Dexter, 2007] Jean-Camille Barros and Arthur J. Dexter. On-Line Identification of Computationally Undemanding Evolving Fuzzy Models. *Fuzzy Sets and Systems*, vol. 158, no. 18, pp. 1997–2012, 2007. (Cited in page 38).

[Bastin and Dochain, 1990] G. Bastin and D. Dochain. *On-line Estimation and Adaptive Control of Bioreactors*. Elsevier, 1990. (Cited in page 137).

[Belchior *et al.*, 2012] Carlos Alberto Coelho Belchior, Rui Alexandre Matos Araújo, and Jorge Afonso Cardoso Landeck. Dissolved Oxygen Control of the Activated Sludge Wastewater Treatment Process Using Stable Adaptive Fuzzy Control. *Computers & Chemical Engineering*, vol. 37, pp. 152–162, February 2012. (Cited in pages xxvi, 64, and 65).

[Bellman and Zadeh, 1970] R. E. Bellman and Lotfi A. Zadeh. Decision-Making in a Fuzzy Environment. *Management Science*, vol. 17, 1970. (Cited in page 24).

[Bello *et al.*, 2014] Oladipupo Bello, Yskandar Hamam, and Karim Djouani. Fuzzy Dynamic Modelling And Predictive Control of a Coagulation Chemical Dosing Unit For Water Treatment Plants. *Journal of Electrical Systems and Information Technology*, vol. 1, no. 2, pp. 129–143, 2014. (Cited in page 79).

[Bemporad *et al.*, 2002] Alberto Bemporad, Manfred Morari, Vivek Dua, and Efstratios N. Pistikopoulos. The Explicit Linear Quadratic Regulator For Constrained Systems. *Automatica*, vol. 38, no. 1, pp. 3–20, January 2002. (Cited in page 33).

[Bemporad *et al.*, 2003] Alberto Bemporad, Francesco Borrelli, and Manfred Morari. Min-max Control of Constrained Uncertain Discrete-time Linear Systems. *IEEE Transactions on Automatic Control*, vol. 48, no. 9, pp. 1600–1606, September 2003. (Cited in page 107).

[Berry and Linoff, 1996] Michael J.Ã. Berry and Gordon S. Linoff. *Data Mining Techniques For Marketing, Sales and Customer Relationship*. Wiley Publish, Inc., USA, 1996. (Cited in page 44).

[Besselmann *et al.*, 2012] Thomas Besselmann, Johan Löfberg, and Manfred Morari. Explicit MPC for LPV Systems: Stability and Optimality. *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2322–2332, 2012. (Cited in page 99).

[Bezdek, 1974] J.C̃. Bezdek. Cluster Validity With Fuzzy Sets. *Cybernet*, vol. 3, no. 3, pp. 58–73, February 1974. (Cited in page 44).

[Bezdek, 1975] J. C. Bezdek. Mathematical models for systematics and taxonomy. *8th Int. Conf. Numerical Taxonomy, G. Estabrook, Ed., Freeman, San Franscisco*, pp. 143–166, 1975. (Cited in page 44).

[Blanchini, 1999] F. Blanchini. Set Invariance in Control. *Automatica*, vol. 35, no. 11, pp. 1747–1767, November 1999. (Cited in page 98).

[Bobál and Chalupa, 2008] Vladimir Bobál and Petr Chalupa. *Self-Tuning Controllers Simulink Library*. Department of Control Theory, Institute of Information Technologies, Faculty of Applied Informatics, Tomas Bata University in Zlín, Zlín, Czech Republic, March 2008.
`http://www.mathworks.com/matlabcentral/fileexchange/8381-stcsl-standard-version`.
(Cited in page 46).

[Bobál *et al.*, 2005] Vladimír Bobál, Josef Böhm, Jaromír Fessl, and Jiří Macháček. *Self-tuning PID Controllers*. Springer London, 2005. (Cited in page 46).

[Cairano *et al.*, 2015] Stefano Di Cairano, Uroš V. Kalabić, and Ilya V. Kolmanovsky. Reference Governor For Network Control Systems Subject to Variable Time-Delay. *Automatica*, vol. 62, pp. 77–86, December 2015. (Cited in page 98).

[Calafiore and Fagiano, 2013] Giuseppe C. Calafiore and Lorenzo Fagiano. Robust Model Predictive Control via Scenario Optimization. *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 219–224, January 2013. (Cited in page 31).

[Camacho and Bordons, 2007] E. F. Camacho and C. Bordons. *Model Predictive Control*. Advanced Textbooks in Control and Signal Processing. Springer-Verlag, London, 2nd ed., July 2007. (Cited in pages xxvi, 78, 83, 86, 88, 90, 92, and 95).

[Cattell, 2012] Raymond Cattell. *The scientific Use Of Factor Analysis in Behavioral And Life Sciences*. Springer Science & Business Media, 2012. (Cited in page 12).

[Chakrabarty *et al.*, 2013] Ankush Chakrabarty, Suvadeep Banerjee, Sayan Maity, and Amitava Chatterjee. Fuzzy Model Predictive Control of Non-linear Processes Using Convolution Models and Foraging Algorithms. *Measurement*, vol. 246, no. 24, pp. 1616–1629, May 2013. (Cited in page 33).

[Chang *et al.*, 2010] Wen-Jer Chang, Cheung-Chieh Ku, and Pei-Hwa Huang. Robust Fuzzy Control for Uncertain Stochastic Time-Delay Takagi-Sugeno Fuzzy Models for Achieving Passivity. *Fuzzy Sets and Systems*, vol. 161, no. 15, pp. 2012–2032, 2010. (Cited in page 38).

[Clarke, 1988] D. W. Clarke. Application of Generalized Predictive Control to Industrial Process. *IEEE Control Systems Magazine*, vol. 8, no. 2, pp. 49–55, 1988. (Cited in page 78).

[Coelho and Herrera, 2007] Leandro dos Santos Coelho and Bruno Meirelles Herrera. Fuzzy Identification Based on a Chaotic Particle Swarm Optimization Approach Applied to Nonlinear Yo-yo Motion System. *IEEE Transactions on Industrial Electronics*, vol. 54, no. 6, pp. 3234–3245, December 2007. (Cited in pages 8, 25, and 38).

[Cutler and Ramaker, 1980] C. R. Cutler and B. L. Ramaker. Dynamic Matrix Control - a Computer Control Algorithm. In: *Proceedings of the Joint Automatic Control Conference*. August 13-15 1980. (Cited in page 30).

[Davies and Walters, 2004] Howard Davies and Peter Walters. Emergent Patterns of Strategy, Environment And Performance In A Transition Economy. *Strategic Management Journal*, vol. 25, no. 4, pp. 347–364, 2004. (Cited in page 12).

[Dayal and MacGregor, 1997] Bhupinder S. Dayal and John F. MacGregor. Recursive Exponentially Weighted PLS and its Applications to Adaptive Control and

Prediction. *Journal of Process Control*, vol. 7, no. 3, pp. 169–179, 1997. (Cited in pages xxvi, 60, 62, 63, 68, 73, 74, and 75).

[Delgado *et al.*, 2001] Myriam Regattieri Delgado, Fernando Von Zuben, and Fernando Gomide. Hierarchical Genetic Fuzzy Systems. *Information Sciences*, vol. 136, no. 1-4, pp. 29–52, 2001. (Cited in pages 25, 38, 39, and 40).

[Delgado *et al.*, 2009] Myriam Regattieri Delgado, Elaine Yassue Nagai, and Lúcia Valéria Ramos de Arruda. A Neuro-Coevolutionary Genetic Fuzzy System to Design Soft Sensors. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, vol. 13, no. 5, pp. 481–495, March 2009. (Cited in pages 27, and 39).

[Dewasmea *et al.*, 2011] L. Dewasmea, B. Srinivasanb, M. Perrierb, and A. Vande Wouwera. Extremum-seeking Algorithm Design For Fed-batch Cultures of Microorganisms with Overflow Metabolism. *Journal of Process Control*, vol. 21, no. 7, pp. 1092–1104, August 2011. (Cited in page 135).

[Diaz *et al.*, 1996] Constantino Diaz, Pierre Dieu, Claude Feuillerat, Philippe Lelong, and Marc Salomé. Simultaneous Adaptive Predictive Control of the Partial Pressures of Dissolved Oxygen (pO 2) and Dissolved Carbon Dioxide (pCO 2) in a Laboratory-scale Bioreactor. *Journal of biotechnology*, vol. 52, no. 2, pp. 135–150, December 1996. (Cited in page 135).

[Dovzan *et al.*, 2015] Dejan Dovzan, Vito Logar, and Igor Skrjanc. Implementation of An Evolving Fuzzy Model (eFuMo) in a Monitoring System For a Waste-water Treatment Process. *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 5, pp. 1761–1776, 2015. (Cited in page 38).

[Dovžan and Škrjanc, 2011] Dejan Dovžan and Igor Škrjanc. Recursive Fuzzy C-Means Clustering for Recursive Fuzzy Identification of Time-Varying Processes. *ISA Transactions*, vol. 50, no. 2, pp. 159–169, 2011. (Cited in pages 42, and 43).

[Eberhart and Shi, 1998] R. C. Eberhart and Y. Shi. Comparison Between Genetic Algorithms and Particle Swarm Optimization. In: *Evolutionary Programming VII*, pp. 611–618. Springer, 1998. (Cited in pages 25, and 39).

[Eisen *et al.*, 1998] Michael B. Eisen, Paul T. Spellman, Patrick O. Brown, and David Botstein. Cluster Analysis And Display of Genome-Wide Expression Patterns. *Proceedings of the National Academy of Sciences*, vol. 95, no. 25, pp. 14863–14868, 1998. (Cited in page 12).

[Eun *et al.*, 1999] Yongsoon Eun, Jung-Ho Kim, Kwangsoo Kim, and Dong-Il Cho. Discrete-time Variable Structure Controller With a Decoupled Disturbance Compensator And its Application to a CNC Servomechanism. *IEEE Transactions on Control Systems Technology*, vol. 7, no. 4, pp. 414–423, July 1999. (Cited in pages xxviii, 125, 141, 142, and 146).

[Evans *et al.*, 2012] Martin Evans, Mark Cannon, and Basil Kouvaritakis. Robust and Stochastic Linear MPC For Systems Subject to Multiplicative Uncertainty. *IFAC Proceedings Volumes*, vol. 45, no. 17, pp. 335–341, 2012. 4th IFAC Conference on Nonlinear Model Predictive Control. (Cited in page 31).

[Feng and Guangren, 2008] Tan Feng and Duan Guangren. Global Stabilizing Controller Design For Linear Time-varying Systems and Its Application on BTT Missiles. *Journal of Systems Engineering and Electronics*, vol. 19, no. 6, pp. 1178–1184, December 2008. (Cited in page 98).

[Feng *et al.*, 2014] Yong Feng, Fengling Han, and Xinghuo Yu. Chattering Free Full-order Sliding-mode Control. *Automatica*, vol. 50, no. 4, pp. 1310–1314, April 2014. (Cited in pages 33, and 100).

[Fisher, 1936] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, September 1936. (Cited in page 14).

[Fleming *et al.*, 2013] James Fleming, Basil Kouvaritakis, and Mark Cannon. Regions of Attraction And Recursive Feasibility in Robust MPC. In: *21st IEEE Mediterranean Conference on Control & Automation (MED 2013)*, pp. 801–806. June 25-28 2013. (Cited in pages 31, and 99).

[Fleming *et al.*, 2015] James Fleming, Basil Kouvaritakis, and Mark Cannon. Robust Tube MPC for Linear Systems With Multiplicative Uncertainty. *IEEE Transactions on Automatic Control*, vol. 60, no. 4, pp. 1087–1092, April 2015. (Cited in pages 31, and 99).

[Gilbert and Kolmanovsky, 2002] Elmer Gilbert and Ilya Kolmanovsky. Nonlinear Tracking Control in the Presence of State and Control Constraints: A Generalized Reference Governor. *Automatica*, vol. 38, no. 12, pp. 2063–2073, December 2002. (Cited in page 98).

[Guang-Yue *et al.*, 2013] Xue Guang-Yue, Ren Xue-Mei, and Xia Yuan-Qing. Multi-rate Sensor Fusion-based Adaptive Discrete Finite-time Synergetic Control For Flexible-joint Mechanical Systems. *Chinese Physics B*, vol. 22, no. 10, p. 100702, April 2013. (Cited in pages 7, and 109).

[Haddad and Chellaboina, 2008] Wassim M. Haddad and VijaySekhar Chellaboina. *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*. Princeton University Press, Princeton, NJ, USA, 2008. (Cited in page 114).

[Hafidi *et al.*, 2008] G. Hafidi, S. Tebbani, D. Dumur, and A. Vande Wouwer. Nonlinear Model Predictive Control Applied to E. Coli Cultures. In: *Proc. 17th World Congress The International Federation of Automatic Control*, IFAC-WC 2008, pp. 14570–14575. July 6-11 2008. (Cited in pages xxx, 135, 136, 137, 138, 139, 144, and 145).

[Hafidi, 2008] Ghizlane Hafidi. *Application de la Commande Prédictive Non-linéaire à la Commande de Culture de Bactéries Escherichia Coli*. PhD Thesis, Faculté des Sciences d'Orsay, Supélec, Université Paris-Sud 11, Paris, France, 2008. (in French). (Cited in pages xxx, 136, 139, 141, and 143).

[Hajiloo, 2016] Amir Hajiloo. *Robust and Multi-Objective Model Predictive Control Design for Nonlinear Systems*. PhD thesis in mechanical engineering, School of Graduate Studies of Concordia University, Concordia University, 2016. In Canada. (Cited in page 100).

[Han *et al.*, 2012] Hong-Gui Han, Jun-Fei Qiao, and Qi-Li Chen. Model Predictive Control of Dissolved Oxygen Concentration Based on a Self-Organizing RBF Neural Network. *Control Engineering Practice*, vol. 20, no. 4, pp. 465–476, April 2012. (Cited in pages 31, and 79).

[Herceg *et al.*, 2013] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari. Multi-Parametric Toolbox 3.0. In: *Proc. of the European Control Conference (ECC 2013)*, pp. 502–510. July 17-19 2013. (Cited in page 125).

[Hua *et al.*, 2013] Hang Hua, Yang Liu, Jianguan Lu, and Junting Zhu. A new Impulsive Synchronization Criterion for T-S Fuzzy Model and its Applications. *Applied Mathematical Modelling*, vol. 37, no. 20-21, pp. 8826–8835, 2013. (Cited in page 38).

[Huang *et al.*, 2012] Guang-Bin Huang, Xiaojing Ding Hongming Zhou, and Rui Zhang. Extreme Learning Machine for Regression and Multiclass Classification. *IEEE Transactions on Systems, And Cybernetics-Part B: Cybernetics*, vol. 42, no. 2, pp. 513–529, April 2012. (Cited in pages 60, 62, 63, 68, 73, and 75).

[Hung and Lin, 2012] Pei-Chia Hung and Sheng-Fuu Lin. The Partial Solutions Consideration Based Self-Adaptive Evolutionary Algorithm: A Learning Structure of Neuro-Fuzzy Networks. *Expert Systems with Applications*, vol. 39, no. 12, pp. 10749–10763, September 2012. (Cited in page 79).

[Izadi *et al.*, 2011] Hojjat A. Izadi, Youmin Zhang, and Brandon W. Gordon. Fault Tolerant Model Predictive Control of Quad-Rotor Helicopters With Actuator Fault Estimation. *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 6343–6348, 2011. (Cited in page 30).

[Jeppsson and Pons, 2004] Ulf Jeppsson and Marie-Noëlle Pons. The COST Benchmark Simulation Model-Current State and Future Perspective. *Control Engineering Practice*, vol. 12, no. 3, pp. 299–304, March 2004. (Cited in page 65).

[Jiang *et al.*, 2015] Weilai Jiang, Chaoyang Dong, and Qing Wang. A Systematic Method of Smooth Switching LPV Controllers Design for a Morphing Aircraft. *Chinese Journal of Aeronautics*, vol. 28, no. 6, pp. 1640–1649, December 2015. (Cited in page 98).

[Jiang, 2009] Zhenhua Jiang. Design of a Nonlinear Power System Stabilizer Using Synergetic Control Theory. *Electric Power Systems Research*, vol. 79, no. 6, pp. 855–862, June 2009. (Cited in page 100).

[Kadlec and Gabrys, 2011] Petr Kadlec and Bogdan Gabrys. Local Learning-Based Adaptive Soft Sensor for Catalyst Activation Prediction. *AIChE Journal*, vol. 57, no. 5, pp. 1288–1301, March 2011. (Cited in pages 60, 62, 63, 73, and 75).

[Kalman and Bertram, 1960] R. E. Kalman and J. E. Bertram. Control System Analysis and Design Via the "Second Method" of Lyapunov: II—Discrete-Time Systems. *Journal of Fluids Engineering*, vol. 82, no. 2, pp. 394–400, June 1960. (Cited in pages 7, and 109).

[Kayadelen, 2011] C. Kayadelen. Soil Liquefaction Modeling by Genetic Expression Programming and Neuro-Fuzzy. *Expert Systems with Applications*, vol. 38, no. 4, pp. 4080–4087, April 2011. (Cited in page 79).

[Kennedy and Eberhart, 1995] J. Kennedy and R. Eberhart. Particle Swarm Optimization. In: *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1945. 27 November- 1 December 1995. (Cited in page 50).

[Kerrigan, 2005] Eric C. Kerrigan. Matlab Invariant Set Toolbox, Version 0.10.5. April 2005. URL `http://www.swmath.org/software/11142`. (Cited in page 125).

[Kerrigan and Maciejowski, 2004] Eric C. Kerrigan and Jan M. Maciejowski. Feedback Min-Max Model Predictive Control Using a Single Linear Program: Robust Stability and the Explicit Solution. *International Journal of Robust and Nonlinear Control*, vol. 14, no. 4, pp. 395–413, March 2004. (Cited in page 107).

[Killian *et al.*, 2014] Michaela Killian, Barbara Mayer, and Martin Kozek. Hierachical Fuzzy MPC Concept For Building Heating Control. *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 12048–12055, 2014. (Cited in page 79).

[Kirubakaran *et al.*, 2014] V. Kirubakaran, T. K. Radhakrishnan, and N. Sivakumaran. Distributed Multiparametric Model Predictive Control Design for a Quadruple Tank Process. *Measurement*, vol. 47, no. January, pp. 841–854, January 2014. (Cited in page 33).

[Kolesnikov, 1994] A. A. Kolesnikov. Synergetic Control Theory [in Russian]. *Moscow, Energoatomizdat*, vol. 344, 1994. (Cited in pages 5, 34, 100, and 108).

[Kolesnikov, 2014] Anatoly A. Kolesnikov. Introduction of Synergetic Control. In: *American Control Conference*, ACC 2014, pp. 3013–3016. June 4-6 2014. (Cited in pages 5, 34, and 108).

[Kolmanovsky and Gilbert, 1998] Ilya Kolmanovsky and Elmer G. Gilbert. Theory and Computation of Disturbance Invariant Sets for Discrete-Time Linear Systems. *Mathematical Problems in Engineering*, vol. 4, no. 4, pp. 317–367, 1998. (Cited in pages 118, and 120).

[Kosko, 1994] B. Kosko. Fuzzy Systems as Universal Approximators. *IEEE Transactions on Computers*, vol. 43, no. 11, pp. 1329–1333, November 1994. (Cited in pages 32, and 78).

[Kothare *et al.*, 1996] Mayuresh V. Kothare, Venkataramanan Balakrishnan, and Manfred Morari. Robust Constrained Model Predictive Control Using Linear Matrix Inequalities. *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996. (Cited in page 31).

[Kouvaritakis *et al.*, 2000] Basil Kouvaritakis, J. Anthony Rossiter, and Jan Schuurmans. Efficient Robust Predictive Control. *IEEE Transactions on Automatic Control*, vol. 45, no. 8, pp. 1545–1549, 2000. (Cited in page 31).

[Kulhavý, 1985] Rudolf Kulhavý. *Probabilistic Identification of Time-Variable Systems with Unknown Model of Parameter Evolution.* PhD thesis, Institute of Information Theory and Automation of Czechoslovak Academy of Sciences, Praha, Czechoslovakia, 1985. (in Czech). (Cited in page 46).

[Kulhavý, 1987] Rudolf Kulhavý. Restricted Exponential Forgetting in Real-Time Identification. *Automatica*, vol. 23, no. 5, pp. 589–600, 1987. (Cited in pages 40, 46, and 68).

[Kwon, 1998] S. H. Kwon. Cluster Validity Index for Fuzzy Clustering. *Electronics Letters*, vol. 34, no. 22, pp. 2176–2177, October 1998. (Cited in page 44).

[Kwong *et al.*, 1994] K.M. Kwong, W.A.and Passino, E.G. Laukonen, and S. Yurkovich. Expert Supervision of Fuzzy Learning Systems With Applications to Reconfigurable Control for Aircraft. In: *Proceedings of the 33rd IEEE Conference on Decision and Control*, pp. 4116–4121. 14-16 December 1994. (Cited in page 29).

[Kwong and Passino, 1996] W.A. Kwong and K.M Passino. Dynamically Focused Fuzzy Learning Control. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, no. 1, pp. 53–74, February 1996. (Cited in page 29).

[Layne and Passino, 1996] Jeffery R. Layne and Kevin M. Passino. Fuzzy Model Reference Learning Control. *Journal of Intelligent and Fuzzy Systems*, vol. 4, no. 1, pp. 33–47, 1996. (Cited in page 29).

[Layne and Passino, 1993] J.R. Layne and K.M. Passino. Fuzzy Model Reference Learning Control for Cargo Ship Steering. *IEEE Control Systems Magazine*, vol. 13, no. 6, pp. 23–34, December 1993. (Cited in page 29).

[Lee *et al.*, 2009] Hoon Lee, Vadim I. Utkin, and Andrey Malinin. Chattering Reduction Using Multiphase Sliding Mode Control. *International Journal of Control*, vol. 82, no. 9, pp. 1720–1737, 2009. (Cited in pages 33, and 100).

[Lee *et al.*, 2003] Ming-Ling Lee, Hung-Yuan Chung, and Fang-Ming Yu. Modeling of Hierarchical Fuzzy Systems. *Fuzzy Sets and Systems*, vol. 138, no. 2, pp. 343–361, 2003. (Cited in pages 26, and 27).

[Lennon and Passino, 1999] William K. Lennon and Kevin M. Passino. Intelligent Control for Brake Systems. *IEEE Transactions on Control Systems Technology*, vol. 7, no. 2, pp. 188–202, 1999. (Cited in page 29).

[Li *et al.*, 2009] Chaoshun Li, Jianzhong Zhou, Xiuqiao Xiang, Qingqing Li, and Xueli An. T-S Fuzzy Model Identification Based on a Novel Fuzzy C-Regression Model Clustering Algorithm. *Engineering Applications of Artificial Intelligence*, vol. 22, no. 4-5, pp. 646–653, June 2009. (Cited in pages xxvi, 60, 62, 63, 68, 73, 74, and 75).

[Li *et al.*, 2014a] Dewei Li, Furong Gao, and Yugeng Xi. Seperated Design of Robust Model Predictive Control For LPV Systems With Periodic Disturbance. *Journal of Process control*, vol. 24, no. 1, pp. 250–260, January 2014. (Cited in pages 31, 32, and 99).

[Li *et al.*, 2012] Yiguo Li, Jiong Shen, Kwang Y. Lee, and Xichui Liu. Offset-free Fuzzy Model Predictive Control of a Boiler-turbine System Based on Genetic

Algorithm. *Simulation Modelling Practice and Theory*, vol. 26, pp. 77–95, August 2012. (Cited in pages 30, and 31).

[Li and Du, 2012] Yimin Li and Yijun Du. Indirect Adaptive Fuzzy Observer and Controller Design Based on Interval Type-2 T-S Fuzzy Model. *Applied Mathematical Modelling*, vol. 36, no. 4, pp. 1558–1569, 2012. (Cited in page 38).

[Li *et al.*, 2014b] Yongming Li, Shaocheng Tong, and Tieshan Li. Adaptive Fuzzy Output-Feedback Control For Output Constrained Nonlinear Systems in the Presence of Input Saturation. *Fuzzy Sets and Systems*, vol. 248, pp. 138–155, August 2014. (Cited in pages 29, and 30).

[Lin and Antsakis, 2009] Hai Lin and P. J. Antsakis. Stability and Stabilizability of Switched Linear Systems: A Survey Of Recent Results. *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 308–322, February 2009. (Cited in page 98).

[Liu and Hsiao, 2012] Chi-Hua Liu and Ming-Ying Hsiao. A finite time synergetic control scheme for robot manipulators. *Computers & Mathematics with Applications*, vol. 64, no. 5, pp. 1163–1169, September 2012. (Cited in pages 7, and 109).

[Liu *et al.*, 2013] Yan-Jun Liu, Shaocheng Tong, and CL Philip Chen. Adaptive Fuzzy Control Via Observer Design For Uncertain Nonlinear Systems With Unmodeled Dynamics. *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 2, pp. 275–288, 2013. (Cited in pages 29, and 30).

[Lo and Chen, 1995] Shih-Che Lo and Yon-Ping Chen. Smooth Sliding-Mode Control For Spacecraft Attitude Tracking Maneuvers. *Journal of Guidance, Control, and Dynamics*, vol. 18, no. 6, pp. 1345–1349, 1995. (Cited in page 33).

[Lucia and Tedesco, 2015] Walter Lucia and Francesco Tedesco. A Networked-Based Receding Horizon Scheme For Constrained LPV Systems. *European Journal of Control*, vol. 25, pp. 69–75, September 2015. (Cited in page 98).

[Maeder *et al.*, 2009] Urban Maeder, Francesco Borrelli, and Manfred Morari. Linear Offset-free Model Predictive Control. *Automatica*, vol. 45, no. 10, pp. 2214–2222, October 2009. (Cited in pages 30, and 31).

[Mamdani, 1974] E. H. Mamdani. Application of Fuzzy Algorithms for Control of Simple Dynamic Plant. *Procedings of IEEE*, vol. 121, no. 12, pp. 1585–1588, December 1974. (Cited in page 24).

[Mamdani and Assilian, 1975] E. H. Mamdani and S. Assilian. An Experiment in Linguistic Synthesis With a Fuzzy Logic Controller. *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, January 1975. (Cited in page 24).

[Mayne and Schroeder, 1997] David Q. Mayne and W. R. Schroeder. Robust Time-optimal Control of Constrained Linear Systems. *Automatica*, vol. 33, no. 12, pp. 2103–2118, December 1997. (Cited in page 98).

[Mayne *et al.*, 2000] David Q. Mayne, James B. Rawlings, Christopher V. Rao, and Pierre O. M. Scokaert. Constrained Model Predictive Control: Stability and Optimality. *Automatica*, vol. 36, no. 6, pp. 789–814, June 2000. (Cited in page 98).

[Mendes *et al.*, 2012] Jérôme Mendes, Francisco Souza, Rui Araújo, and Nuno Gonçalves. Genetic Fuzzy System for Data-Driven Soft Sensors Design. *Applied Soft Computing*, vol. 12, no. 10, pp. 3237–3245, October 2012. (Cited in pages 27, 39, 40, 60, 62, 63, 68, 73, and 75).

[Mendes *et al.*, 2013] Jérôme Mendes, Rui Araújo, and Francisco Souza. Adaptive Fuzzy Identification and Predictive Control for Industrial Processes. *Expert Systems with Applications*, vol. 40, no. 17, pp. 6964–6975, 2013. (Cited in pages 39, 40, 42, 46, 81, and 82).

[Mendes *et al.*, 2014] Jérôme Mendes, Rui Araújo, Tiago Matias, Ricardo Seco, and Carlos Belchior. Automatic Extraction of the Fuzzy Control System by Hierarchical Genetic Algorithm. *Engineering Applications of Artificial Intelligence*, vol. 29, pp. 70–78, 2014. (Cited in pages 28, 39, 40, 65, and 88).

[Mendes *et al.*, 2016] Jérôme Mendes, Luís Osório, and Rui Araújo. Self-Tuning PID Controllers In Pursuit of Plug and Play Capacity: Comparison and Tests. *Control Engineering Practice*, December 2016. (submitted). (Cited in page 71).

[Mendes, 2014] Jérôme Amaro Pires Mendes. *Computational Intelligence Methodologies for Control of Industrial Processes*. Ph.D. thesis, Universidade de Coimbra, 2014. (Cited in pages 8, and 79).

[Miyamoto, 2012] Sadaaki Miyamoto. *Fuzzy Sets In Information Retrieval And Cluster Analysis*. Springer Science & Business Media, 2012. (Cited in page 12).

[Morningred *et al.*, 1992] J. Duane Morningred, Bradley E. Paden, Dale E. Seborg, and Duncan A. Mellichamp. An Adaptive Nonlinear Predictive Controller. *Chemical Engineering Science*, vol. 47, no. 4, pp. 755–762, 1992. (Cited in page 126).

[Moudgal *et al.*, 1995] V.G. Moudgal, W.A. Kwong, K.M. Passino, and S Yurkovich. Fuzzy Learning Control For a Flexible-Link Robot. *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 2, pp. 199–210, May 1995. (Cited in page 29).

[Muñoz-Carpintero *et al.*, 2015] Diego Muñoz-Carpintero, Mark Cannon, and Basil Kouvaritakis. Robust MPC strategy With Optimized Polytopic Dynamics For Linear Systems With Additive and Multiplicative Uncertainty. *Systems & Control Letters*, vol. 81, pp. 34–41, July 2015. (Cited in pages 31, and 98).

[Navalkar and van Wingerden, 2015] Sachin T. Navalkar and Jan-Willem van Wingerden. Iterative Feedback Tuning of an LPV Feedforward Controller for Wind Turbine Load Alleviation. *IFAC-PapersOnLine*, vol. 48, no. 26, pp. 207–2122, 2015. (Cited in page 98).

[Nelles, 2000] Oliver Nelles. *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*. Springer, 2000. (Cited in pages 24, and 29).

[Nelles, 2001] Oliver Nelles. *Nonlinear System Identification: from Classical to Neural Networks and Fuzzy Models*. Springer, 2001. (Cited in pages 60, 62, 63, 68, 73, and 75).

[Nusawardhana *et al.*, 2007] Nusawardhana, S. H. Żak, and W. A. Crossley. Nonlinear Synergetic Optimal Controllers. *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 4, pp. 1134–1147, July-August 2007. (Cited in page 109).

[Olfati-Saber and Murray, 2002] Reza Olfati-Saber and Richard M. Murray. Distributed Cooperative Control of Multiple Vehicle Formations Using Structural Potential Functions. In: *Proc. 15th IFAC World Congress*, pp. 346–352. July 21-26 2002. (Cited in pages 7, and 109).

[Pan *et al.*, 2015] Huihui Pan, Weichao Sun, Huijun Gao, Tasawar Hayat, and Fuad Alsaadi. Nonlinear Tracking Control Based on Extended State Observer For Vehicle Active Suspensions With Performance Constraints. *Mechatronics*, vol. 30, pp. 363–370, September 2015. (Cited in page 98).

[Pannocchia, 2004] Gabriele Pannocchia. Robust Model Predictive Control with Guaranteed Setpoint Tracking. *Journal of Process Control*, vol. 14, no. 8, pp. 927–937, December 2004. (Cited in pages xxvii, xxx, 32, 98, 99, 125, 126, 132, 133, 134, 135, and 155).

[Pannocchia and Kerrigan, 2005] Gabriele Pannocchia and Eric C. Kerrigan. Offset-free Receding Horizon Control of Constrained Linear Systems. *AIChE Journal*, vol. 51, no. 12, pp. 3134–3146, August 2005. (Cited in pages 107, and 118).

[Passino *et al.*, 1995] K.M. Passino, E.G. Laukonen, and S. Yurkovitch. Expert Supervision of Fuzzy Learning Systems For Fault Tolerant Aircraft Control. *Proceedings of the IEEE*, vol. 83, no. 3, pp. 466–483, March 1995. (Cited in page 29).

[Procyk and Mamdani, 1978] T. J. Procyk and E. H. Mamdani. A Linguistic Self-Organising Process Controller. *Automatica*, vol. 15, no. 1, January 1978. (Cited in page 29).

[Qi *et al.*, 2015] Rongbin Qi, Hua Mei, Chao Chen, and Feng Qian. A Fast MPC Algorithm For Reducing Computation Burden of MIMO. *Chinese Journal of Chemical Engineering*, vol. 23, no. 12, p. 2087–2091, December 2015. (Cited in page 33).

[Qin and Badgwell, 1997] S. Joe Qin and Thomas A. Badgwell. An Overview of Industrial Model Predictive Control Technology. In: *Chemical Process Control-V: Assessment and New Directions for Research: Proceedings of the Fifth International Conference on Chemical Process Control, Tahoe City, California, January 7-12, 1996*, pp. 232–256. American Institute of Chemical Engineers, 1997. (Cited in page 4).

[Qin and Badgwell, 2000] S. Joe Qin and Thomas A. Badgwell. An Overview of Nonlinear Model Predictive Control Applications. In: *Nonlinear Model Predictive Control*, pp. 369–392. Springer-Verlag, 2000. (Cited in pages xxv, and 5).

[Qin and Badgwell, 2003] S. Joe Qin and Thomas A. Badgwell. A Survey of Industrial Model Predictive Control Technology. *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, July 2003. (Cited in page 4).

[Rajua *et al.*, 1991] G. V. S. Rajua, Jun Zhou, and Roger A. Kisner. Hierarchical Fuzzy Control. *International journal of control*, vol. 54, no. 5, pp. 1201–1216, 1991. (Cited in page 27).

[Rastegar and Araújo, 2013] Saeid Rastegar and Rui Araújo. A Robust Algorithm for A Class of Optimal Control Problems Subject to Regional Stability Constraints and Disturbances. *Measurement*, vol. 46, no. 6, pp. 1773–1780, July 2013. (Cited in page 5).

[Rastegar *et al.*, 2016a] Saeid Rastegar, Rui Araújo, Alireza Emami, and Abdelhamid Iratni. A New Robust Control Scheme for LTV Systems Using Output Integral Discrete Synergetic Control Theory. In: *Proceedings of the 12th Portuguese Conference on Automatic Control*, pp. 117–127. September 14-16 2016. (Cited in pages 9, and 109).

[Rastegar *et al.*, 2016b] Saeid Rastegar, Rui Araújo, and Jérôme Mendes. A New Approach for Online T-S Fuzzy Identification and Model Predictive Control of Nonlinear Systems. *Journal of Vibration and Control*, vol. 22, no. 7, pp. 1820–1837, November 2016. (Cited in pages 8, 9, 29, 47, 81, 82, and 86).

[Rastegar *et al.*, 2017a] Saeid Rastegar, Rui Araújo, and Abdelhamid Iratni. A Robust Output Integral Discrete-Time Synergetic Control Design Applied to E. Coli Cultures. *Journal of Process Control*, 2017. (submitted). (Cited in pages 9, 34, and 121).

[Rastegar *et al.*, 2017b] Saeid Rastegar, Rui Araújo, and Jérôme Mendes. Online Identification of Takagi-Sugeno Fuzzy Models Based on Self-Adaptive Hierarchical Particle Swarm Optimization Algorithm. *Applied Mathematical Modelling*, vol. 45, pp. 606–620, January 2017. (Cited in pages 8, and 51).

[Rastegar *et al.*, 2017c] Saeid Rastegar, Rui Araújo, and Jalil Sadati. Robust Synergetic Control Design Under Input and State Constraints. *International Journal of Control*, 2017. (accepted). (Cited in pages 9, 34, 71, 109, and 112).

[Rastegar *et al.*, 2017d] Saeid Rastegar, Rui Araújo, Jalil Sadati, and Jérôme Mendes. A Novel Robust Control Scheme for LTV Systems Using Output Integral Discrete-Time Synergetic Control Theory. *European Journal of Control*, vol. 34, pp. 39–48, March 2017. (Cited in pages 9, 34, 71, and 109).

[Ratnaweera *et al.*, 2004] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson. Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients. *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240–255, June 2004. (Cited in page 57).

[Richalet *et al.*, 1978] J. Richalet, A. Rault, J.L. Testud, and J. Papon. Model Predictive Heuristic Control: Applications to Industrial Processes. *Automatica*, vol. 14, no. 5, pp. 413–428, September 1978. (Cited in page 30).

[Riverso *et al.*, 2014] Stefano Riverso, Marcello Farina, and Giancarlo Ferrari-Trecate. Plug-and-Play Model Predictive Control Based on Robust Control Invariant Sets. *Automatica*, vol. 50, no. 8, pp. 2179–2186, August 2014. (Cited in page 98).

[Rocha, 2003] Isabel Cristina Almeida Pereira Rocha. *Model-Based Strategies for Computer-Aided Operation of a Recombinant* E. Coli *Fermentation*. PhD Thesis, Department of Biological Engineering, School of Engineering, University of Minho, Braga, Portugal, 2003. (Cited in pages xxx, 136, 139, 141, and 143).

[Roopaei and Zolghadri, 2009] M. Roopaei and M. Jahromi Zolghadri. Chattering-Free Fuzzy Sliding Mode Control in MIMO Uncertain Systems. *Nonlinear Analysis: Theory, Methods & Applications*, vol. 71, no. 10, pp. 4430–4437, 2009. (Cited in page 33).

[Rotondo *et al.*, 2014] Damiano Rotondo, Fatiha Nejjari, and Vicenç Puig. Robust State-Feedback Control of Uncertain LPV Systems: An LMI-Based Approach. *Journal of the Franklin Institute*, vol. 351, no. 5, pp. 2781–2803, May 2014. (Cited in page 99).

[Salahshoor *et al.*, 2012] Karim Salahshoor, Morteza Hamzehnejad, and Sepide Zakeri. Online Affine Model Identification of Nonlinear Processes Using a New adaptive Neuro-Fuzzy Approach. *Applied Mathematical Modelling*, vol. 36, no. 11, pp. 5534–5554, 2012. (Cited in page 38).

[Santi *et al.*, 2003] Enrico Santi, Antonello Monti, Donghong Li, Karthik Proddutur, and Roger A. Dougal. Synergetic Control for DC-DC Boost Converter: Implementation Options. *IEEE Transactions on Industry Applications*, vol. 39, no. 6, pp. 1803–1813, December 2003. (Cited in page 100).

[Santi *et al.*, 2004] Enrico Santi, Antonello Monti, D. Li, K. Proddutur, and R. A Dougal. Synergetic control for power electronics applications: a comparison with the sliding mode approach. *Journal of Circuits, Systems, and Computers*, vol. 13, no. 04, pp. 737–760, Aguest 2004. (Cited in page 100).

[Santos *et al.*, 2012] L.O. Santos, L. Dewasmec, D. Coutinhob, and A. Vande Wouwerc. Nonlinear Model Predictive Control of Fed-batch Cultures of Microorganisms Exhibiting Overflow Metabolism: Assessment and Robustness. *Computers and Chemical Engineering*, vol. 39, pp. 143–151, December 2012. (Cited in pages 135, 138, and 140).

[Sastry, 1999] Shankar Sastry. *Nonlinear Systems: Analysis, Stability, and Control*. Springer, New York, NY, USA, 1999. (Cited in pages 7, and 109).

[Shi and Eberhart, 1998] Y. Shi and R. C. Eberhart. A Modified Particle Swarm Optimizer. In: *Proceedings of the IEEE International Conference on Evolutionary Computation,USA*, pp. 69–73. 4-9 May 1998. (Cited in pages 39, 51, and 57).

[Shuzhi *et al.*, 2012] Gao Shuzhi, Dou Xing, and Gao Xianwen. The Temperature System Identification of the PVC Stripper Tower Top Based on PSO-FCM Optimized T-S Model. In: *Proceedings of the IEEE International Conference on Control and Decision (CCD)*, pp. 2529–2532. 10-13 December 2012. (Cited in pages 8, 25, and 38).

[Sinha and Rózsa, 1976] N. K. Sinha and P. Rózsa. Some Canonical Forms for Linear Multivariable Systems. *International Journal of Control*, vol. 23, no. 6, pp. 865–883, 1976. (Cited in pages 106, and 121).

[Škrjanc, 2009] Igor Škrjanc. Confidence Interval of Fuzzy Models: An Example Using a Waste-water Treatment Plant. *Chemometrics and Intelligent Laboratory Systems*, vol. 96, no. 2, pp. 182–187, 2009. (Cited in page 38).

[Smoczek, 2015] Jaroslaw Smoczek. Experimental Verification of a GPC-LPV Method With RLS and P1-TS Fuzzy-based Estimation For Limiting the Transient and Residual Vibration of a Crane System. *Mechanical Systems and Signal Processing*, vol. 62-63, pp. 324–340, October 2015. (Cited in pages 78, and 79).

[Souza and Araújo, 2014] Francisco Souza and Rui Araújo. Online Mixture of Univariate Linear Regression Models for Adaptive Soft Sensors. *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 937–945, 2014. (Cited in page 61).

[Souza *et al.*, 2010] Francisco Souza, Pedro Santos, and Rui Araújo. Variable and Delay Selection Using Neural Networks and Mutual Information for Data-driven Soft Sensors. In: *Proc. IEEE Conference on Emerging Technologies and Factory Automation (ETFA), 2010*, pp. 1–8. 13-16 September 2010. (Cited in pages 31, 49, and 79).

[Souza *et al.*, 2013] Francisco Souza, Rui Araújo, Tiago Matias, and Jérôme Mendes. A Multilayer-Perceptron Based Method For Variable Selection In Soft Sensor Design. *Journal of Process Control*, vol. 23, no. 10, pp. 1371–1378, 2013. (Cited in page 61).

[Spooner *et al.*, 2002] Jeffrey T. Spooner, Manfredi Maggiore, and Raúl Ordó nez. *Stable Adaptive Control and Estimation for Nonlinear Systems.* John Wiley Sons, Inc, 2002. (Cited in page 29).

[Takagi and Sugeno, 1985] Tomohiro Takagi and Michio Sugeno. Fuzzy Identification of Systems and Its Applications to Modeling and Control. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15, no. 1, pp. 116–132, February 1985. (Cited in pages 25, 38, and 41).

[Tang and Sun, 2005] Yuangang Tang and Fuchun Sun. Improved Validation Index for Fuzzy Clustering. In: *American Control Conference*, pp. 1120–1125. 8-10 June 2005. (Cited in page 44).

[Tham *et al.*, 1991] M. T. Tham, F. Vagi, A. J. Morris, and R. K. Wood. Multivariable and Multirate Self-Tuning Control: a Distillation Column Case Study. *IEE Proceedings Part D*, vol. 138, no. 1, pp. 9–24, 1991. (Cited in page 78).

[Tóth, 2010] Roland Tóth. *Modeling and Identification of Linear Parameter-Varying Systems*. Springer-Verlag, Berlin, Germany, 2010. (Cited in pages 106, and 121).

[Unal *et al.*, 2003] Yurdanur Unal, Tayfun Kindap, and Mehmet Karaca. Redefining The Climate Zones Of Turkey Using Cluster Analysis. *International journal of climatology*, vol. 23, no. 9, pp. 1045–1055, 2003. (Cited in page 12).

[Vizer *et al.*, 2015] Daniel Vizer, Guillaume Mercere, and Edouard Laroche. Gray-box LPV Model Identification of A 2-DoF Surgical Robotic Manipulator by Using An H-infinity Norm-Based Local Approach. *IFAC-PapersOnLine*, vol. 48, no. 26, pp. 79–84, 2015. (Cited in page 98).

[Wan and Kothare, 2002] Zhaoyang Wan and Mayuresh V. Kothare. Robust Output Feedback Model Predictive Control Using Off-line Linear Matrix Inequalities. *Journal of Process Control*, vol. 12, no. 7, pp. 763–774, October 2002. (Cited in pages xxx, 98, 126, and 127).

[Wan and Kothare, 2003] Zhaoyang Wan and Mayuresh V. Kothare. An Efficient Off-line Formulation of Robust Model Predictive Control Using Linear Matrix Inequalities. *Automatica*, vol. 39, no. 5, pp. 837–846, May 2003. (Cited in page 98).

[Wang, 1992] Li-Xin Wang. Stable Adaptive Fuzzy Control of Nonlinear Systems. In: *Proceedings of the 31st IEEE Conference on Decision and Control*, pp. 2511–2516. 16-18 December 1992. (Cited in page 29).

[Wang, 1996] Li-Xin Wang. Stable Adaptive Fuzzy Controllers With Application to Inverted Pendulum Tracking. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, no. 5, pp. 677–691, October 1996. (Cited in pages 29, and 54).

[Wang, 1997a] Li-Xin Wang. *A Course in Fuzzy Systems and Control*. Prentice-Hall, Inc., 1997. (Cited in pages 17, 19, 20, and 28).

[Wang, 1997b] Li-Xin Wang. *A Course in Fuzzy Systems and Control*. Prentice-Hall, Inc., 1997. (Cited in page 41).

[Wang and Mendel, 1992] Li-Xin Wang and J.M. Mendel. Fuzzy Basis Functions, Universal Approximation, and Orthogonal Least-Squares Learning. *IEEE Trans-*

actions on Neural Networks, vol. 3, no. 5, pp. 807–814, September 1992. (Cited in pages 32, and 78).

[Wang *et al.*, 2016] Ning Wang, Jing-Chao Sun, and Yan-Cheng Liu. Direct Adaptive Self-Structuring Fuzzy Control With Interpretable Fuzzy Rules For a Class of Nonlinear Uncertain Systems. *Neurocomputing*, vol. 173, pp. 1640–1645, 2016. (Cited in page 28).

[Wang *et al.*, 2013] Yizhou Wang, Wenjie Chen, Masayoshi Tomizuka, and Badr N. Alsuwaidan. Model Predictive Sliding Mode Control For Constrain Satisfaction and Robustness. In: *ASME 2013 Dynamic Systems and Control Conference*, pp. V003T44A005–V003T44A005. October 21-23 2013. (Cited in pages 33, and 100).

[Wassink *et al.*, 2005] Matthijs Groot Wassink, Marc van de Wal, Carsten Scherer, and Okko Bosgra. LPV Control For a Wafer Stage: Beyond the Theoretical Solution. *Control Engineering Practice*, vol. 13, no. 2, pp. 231–245, February 2005. (Cited in page 98).

[Weinschenk *et al.*, 2003] Jeffrey J. Weinschenk, William E. Combs, and Robert J. Marks. Avoidance of Rule Explosion by Mapping Fuzzy Systems to a Union Rule Configuration. In: *The 12th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2003)*, pp. 43–48. 25-28 May 2003. (Cited in page 27).

[Weinschenk *et al.*, 2004] Jeffrey J. Weinschenk, Robert J. Marks, and William E. Combs. On the Use of Fourier Methods in URC Fuzzy System Design. In: *2004 IEEE International Conference on Fuzzy Systems*, pp. 911–916. 25-29 July 2004. (Cited in page 27).

[Windham, 1982] M. P. Windham. Cluster Validity for the Fuzzy c-Means Clustering Algorithm. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol. PAMI-4, no. 4, pp. 357–363, July 1982. (Cited in page 44).

[Wu, 2012] Kuo-Lung Wu. Analysis of Parameters Selections For Fuzzy C-Mean. *Pattern Recognition*, vol. 45, no. 1, pp. 407–415, January 2012. (Cited in pages 45, and 47).

[Wu *et al.*, 2012] Min Wu, Chunsheng Wang, Weihua Cao, Xuzhi Lai, and Xin Chen. Design and Application of Generalized Predictive Control Strategy With

Closed-Loop Identification for Burn-Through Point in Dintering Process. *Control Engineering Practice*, vol. 20, no. 10, pp. 1065–1074, October 2012. (Cited in pages 78, and 79).

[Xie and Beni, 1991] X. Xie and G. Beni. A Validity Measure for Fuzzy Clustering. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol. 13, no. 8, pp. 841–847, August 1991. (Cited in page 44).

[Xu and Li, 2012] Qingsong Xu and Yangmin Li. Micro-/nanopositioning Using Model Predictive Output Integral Discrete Sliding Mode Control. *IEEE Transactions on Industrial Electronics*, vol. 59, no. 2, pp. 1161–1170, February 2012. (Cited in pages 111, and 122).

[Yao *et al.*, 2014] Yanqing Yao, Jusheng Mi, and Zhoujun Li. A Novel Variable Precision (θ, σ)-Fuzzy Rough Set Model Based on Fuzzy Granules. *Fuzzy Sets and Systems*, vol. 236, pp. 58–72, February 2014. (Cited in page 39).

[Ying, 1997] H. Ying. General MISO Takagi-Sugeno Fuzzy Systems with Simplified Linear Rule Consequent as Universal Approximators for Control and Modeling Applications. *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, pp. 1335–1340, 1997. (Cited in page 40).

[Yu *et al.*, 2004] Jian Yu, Qiansheng Cheng, and Houkuan Huang. Analysis of the Weighting Exponent in the FCM. *IEEE Trans Syst Man Cybern B*, vol. 34, no. 1, pp. 634–639, February 2004. (Cited in pages 45, and 47).

[Zadeh, 1965] Lotfi A. Zadeh. Fuzzy Sets. *Information and Control*, vol. 8, no. 3, pp. 338–353, June 1965. (Cited in page 24).

[Zadeh, 1968] Lotfi A. Zadeh. Fuzzy Algorithms. *Information and Control*, vol. 12, no. 2, pp. 94–102, 1968. (Cited in page 24).

[Zadeh, 1971] Lotfi A. Zadeh. Similarity Relations and Fuzzy Orderings. *Information Science*, vol. 3, no. 2, pp. 177–200, 1971. (Cited in page 24).

[Zadeh, 1973] Lotfi A. Zadeh. Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, pp. 28–44, January 1973. (Cited in page 24).

[Zhang *et al.*, 2009] Tiejun Zhang, Gang Feng, and Xiao-Jun Zeng. Output Tracking of Constrained Nonlinear Processes With Offset-free Input-to-state Stable Fuzzy Predictive Control. *Automatica*, vol. 45, no. 4, pp. 900–909, April 2009. (Cited in page 31).

[Ziegler and Nichols, 1942] John G. Ziegler and B. Nathaniel Nichols. Optimum Settings For Automatic Controllers. *Trans. ASME*, vol. 64, no. 11, 1942. (Cited in pages 89, and 92).

[Zumberge and Passino, 1998] Jon Zumberge and Kevin M. Passino. A Case Study in Intelligent vs. Conventional Control For a Process Control Experiment. *Control Engineering Practice*, vol. 6, no. 9, pp. 1055–1075, September 1998. (Cited in page 29).


FCT
Fundação para a Ciência e a Tecnologia
MINISTÉRIO DA EDUCAÇÃO E CIÊNCIA


mais
CENTRO
Programa Operacional Regional do Centro


adi
agência de inovação


POPH
PROGRAMA OPERACIONAL POTENCIAL HUMANO


QREN
QUADRO
DE REFERÊNCIA
ESTRATÉGICO
NACIONAL
PORTUGAL 2007-2013


GOVERNO DA REPÚBLICA
PORTUGUESA


UNIÃO EUROPEIA
Fundo Social Europeu