Maude Josée Blondin
Panos M. Pardalos
Javier Sanchis Sáez   *Editors*

Computational Intelligence and Optimization Methods for Control Engineering

Springer

Springer Optimization and Its Applications

Volume 150

Aims and Scope
Optimization has been expanding in all directions at an astonishing rate during the last few decades. New algorithmic and theoretical techniques have been developed, the diffusion into other disciplines has proceeded at a rapid pace, and our knowledge of all aspects of the field has grown even more profound. At the same time, one of the most striking trends in optimization is the constantly increasing emphasis on the interdisciplinary nature of the field. Optimization has been a basic tool in all areas of applied mathematics, engineering, medicine, economics and other sciences.

The series **Springer Optimization and Its Applications** aims to publish state-of-the-art expository works (monographs, contributed volumes, textbooks) that focus on algorithms for solving optimization problems and also study applications involving such problems. Some of the topics covered include nonlinear optimization (convex and nonconvex), network flow problems, stochastic optimization, optimal control, discrete optimization, multi-objective programming, description of software packages, approximation techniques and heuristic approaches.

Volumes from this series are indexed by Web of Science, zbMATH, Mathematical Reviews, and SCOPUS.

More information about this series at http://www.springer.com/series/7393

Maude Josée Blondin · Panos M. Pardalos ·
Javier Sanchis Sáez

Editors

Computational Intelligence and Optimization Methods for Control Engineering

Editors
Maude Josée Blondin
Department of Mechanical
and Aerospace Engineering
University of Florida
Gainesville, FL, USA

Panos M. Pardalos
Department of Industrial
and Systems Engineering
University of Florida
Gainesville, FL, USA

Javier Sanchis Sáez
Institute of Automation
and Industrial Computer
Polytechnic University of Valencia
Valencia, Spain

Foreword

*Science is a way of trying not to fool yourself. The principle
is that you must not fool yourself, and you are the easiest
person to fool.*
—Richard Feynman

On the one hand, computational intelligence is a vast and maturing field, which
has been applied to a large scope of applications. Undeniably, many real-life
problems cannot be translated into a language for computers to process it easily.
Computational intelligence is a set of nature-inspired computational methodolo-
gies and approaches to address complex real-world problems to which typical
mathematical modeling can be useless. Consequently, computational intelligence
provides solutions for such problems. On the other hand, optimization is in a
pragmatic sense the act of obtaining the best results under given circumstances.
Essentially, optimization goes hand-in-hand with high-speed digital computers
making possible the implementation of complex optimization methods and
stimulating further research on newer methods. Combining computational intel-
ligence and optimization methodologies to be applied to control engineering
problems is a natural advancement of this research field, and its diffusion to a
wide range of disciplines is one of the most recent and striking trends. In a close
future, developments in this area will play an essential position in a wide range of
high-performance systems, from simple household appliances to complex trans-
portation systems. The time has come to propose an emerging expertise as a
conjunction of computational intelligence and optimization to reach better real-life
implementation on control engineering applications, more than to open deep
discussion on theoretical or/and conceptual development on isolated domains. The
present volume of Springer Optimization and Its Applications series by
Maude J. Blondin, Panos M. Pardalos, and Javier Sanchis Sáez is devoted to the
expansion of this expertise. The editorial team represents well the ongoing trend
of this emerging expertise. Maude J. Blondin, postdoctoral researcher at the
University of Florida, is the youngest but very active in the field. I have the
pleasure to meet her several times in international conferences and I follow her
research developments actively. She is very well assisted by two eminent pro-
fessors: P. Pardalos, distinguished professor in industrial and systems engineering

at the University of Florida; and Javier Sanchis Sáez, professor in the department
of systems engineering and control of Universitat Politécnica de València, Spain,
inevitable reference names in this research field. The volume presents the role of
computational intelligence- and optimization methods-based tools play in a large
variety of control engineering applications. Starting with a comprehensive over-
view of control engineering, the volume focuses on real applications such as
transport optimization including autonomous vehicles and hybrid electric buses
case studies; energy optimization including hybrid electric buses, load monitoring
in residences, power estimation, microgrids, and renewable systems case studies;
manufacturing optimization including smart composite structures and semicon-
ductor case studies; control optimization including complex systems, parameter
tuning, cyber-physical- human systems, and complex biological systems case
studies. As several applications with increasing complexity are presented, this
volume is directed toward entry level scientists, engineers, and technicians in
industry and governmental agencies as well in academia. The multidisciplinary
and vast application real-life problems that are proposed in this volume demon-
strate the huge potential of *Computational Intelligence and Optimization Methods
for Control Engineering*. It will be an excellent textbook support to the reader to
extrapolate this new trend expertise to his own problems.

April 2019   João Pedro F. Trovão
                                                                           Canada Research Chairholder
                                                                                  Faculty of Engineering
                                                                           University of Sherbrooke
                                                                                  Sherbrooke, Canada

Preface

Nowadays, control engineering is an important field of research since it is part of almost all engineering activities and systems. The proportional–integral–derivative (PID) controller proposed in the beginning of the twentieth century is the most used controller to this day, mostly in industrial applications. However, technological advances and technology affordability have enabled the deployment of new control systems, in particular intelligent systems. Intelligent systems, which include automatic control, cover a broad range of applications for which PID control has practical limitations. As a result, control theory techniques related to computational intelligence and soft computing approaches have been expanding at a high speed over the last decades. It is still a current research trend. Indeed, there are hundreds of publications on this topic that may cause even someone knowledgeable in the subject to lose track of all novelties. Therefore, the main purpose of the book is to present some recent and important developments related to computational intelligence and optimization methods in control, with attention to position the control techniques into the literature through state-of-the-art surveys. Future directions and research perspectives on the subject are clearly presented. Consequently, this book is designed to give the reader a better understanding of different challenges faced in control engineering and to provide insights to enable the development of new techniques. In addition, the book empowers the reader to determine the appropriate control method for a specific application.

The reading of this book is not intended to be linear, i.e., each chapter stands on its own. However, the chapter order follows a logical flow of ideas that may help a novice in control to understand better the presented concepts.

The first chapter of the book introduces the reader to control engineering. Traditional control techniques along with their limitations to highlight the importance of new control methods are presented. Modern control techniques such as optimal control and adaptive control are also exposed. However, this chapter focuses mainly on intelligent control techniques, which include metaheuristics control tuning, fuzzy logic control, neural networks, and multi-agent systems. Perspective and new trends of research are exposed for each presented control technique as well as for control systems in general. The second chapter presents

some of the main metaheuristics that are used in control, i.e., taboo search, simulated annealing, tunneling algorithms, GRASP methods, genetic algorithms and evolutionary algorithms, ant colony optimization, particle swarm optimization, and multi-objective optimization. In the same direction, Chapter 3 presents controller tuning with metaheuristics on canonical benchmark control problems. This chapter provides good insight and explains in detail how to apply metaheuristics to controller tuning. Chapter 4 explains fuzzy logic and neuro-fuzzy logic techniques with applicative examples in smart structures. Chapters 5 and 6 deal specifically with neural networks for system applications in desalination industry and in blood glucose control devices. Distributed optimization and cooperative control, which is related to multi-agent concept, are presented in Chapters 7 and 8 for the digital grid and microgrids, respectively. Chapters 9–14 propose novel control techniques with illustrative examples of applications in power systems, load monitoring systems, autonomous vehicles, controllers for semiconductor manufacturing, and hybrid electric buses. Most of the control problems presented in this book deal with nonlinearities and their limitations. Perturbation management, robustness, stability, and dynamic performances are also considered to assure practice validity.

In summary, the contributed book covers theoretical aspects of control engineering along with practical applications.

Gainesville, FL, USA   Maude Josée Blondin
Gainesville, FL, USA   Panos M. Pardalos
Valencia, Spain   Javier Sanchis Sáez
April 2019

Acknowledgements

Contents

Contributors

O. Tolga Altinoz Department of Electrical and Electronics Engineering, Ankara University, Ankara, Turkey

Samuel Oludare Bamgbose Department of Electrical and Computer Engineering and CREDIT Center, Prairie View A&M University, Prairie View, TX, USA

Rahul Bhadani Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ, USA

Maude Josée Blondin University of Florida, Gainesville, Florida, USA

Pierre Borne Centre de recherche en informatique, signal et automatique deLille. Ecole Centrale de Lille, Villeneuve d'Ascq Cédex, France

Philipp Braun School of Electrical Engineering and Computing, University of Newcastle, Newcastle, Australia

Pedro Cabrera Department of Mechanical Engineering, University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Canary Islands, Spain

Haritza Camblong University of the Basque Country, Donostia San Sebastian, Gipuzkoa, Spain;
ESTIA Research, Bidart, France

José A. Carta Department of Mechanical Engineering, University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Canary Islands, Spain

Harold R. Chamorro KTH Royal Institute of Technology, Stockholm, Sweden

Chen Fu Chien National Tsing Hua University, Hsinchu, Taiwan

Maria Laura Delle Monache University of Grenoble Alpes, Inria, CNRS, Grenoble INP, GIPSA-Lab, Grenoble, France

Mahdi Fathi Mississippi State University, Starkville, MS, USA

Haizea Gaztañaga IKERLAN Technology Research Centre, Energy Storage and Management Area, Arrasate, Gipuzkoa, Spain

George Georgiou Department of Informatics and Telecommunications, University of Ioannina, Arta, Greece

Amira Gharbi Centre de recherche en informatique, signal et automatique deLille. Ecole Centrale de Lille, Villeneuve d'Ascq Cédex, France

Christos Gogos Department of Informatics and Telecommunications, University of Ioannina, Arta, Greece

Francisco Gonzalez-Longatt Loughborough University, Loughborough, UK

Azwirman Gusrialdi Tampere University, Tampere, Finland

Victor Isaac Herrera Facultad de Informática y Electrónica, Escuela Superior Politécnica de Chimborazo, Riobamba, Ecuador

Marzieh Khakifirooz Tecnológico de Monterrey, Monterrey, Mexico

Xiangfang Li Department of Electrical and Computer Engineering and CREDIT Center, Prairie View A&M University, Prairie View, TX, USA

Thibault Liard University of Grenoble Alpes, Inria, CNRS, Grenoble INP, GIPSA-Lab, Grenoble, France

Jon Ander López IKERLAN Technology Research Centre, Energy Storage and Management Area, Arrasate, Gipuzkoa, Spain

Aitor Milo IKERLAN Technology Research Centre, Energy Storage and Management Area, Arrasate, Gipuzkoa, Spain

Camilo A. Ordonez Grupo Energia Bogota, Bogotá, Colombia

Panos M. Pardalos University of Florida, Gainesville, FL, USA

Jimmy C.-H. Peng National University of Singapore, Singapore, Singapore

Benedetto Piccoli Department of Mathematics, University of Rutgers, Camden, NJ, USA

Lijun Qian Department of Electrical and Computer Engineering and CREDIT Center, Prairie View A&M University, Prairie View, TX, USA

Zhihua Qu University of Central Florida, Orlando, USA

Anaïs Rat Department of Mathematics, University of Rutgers, Camden, NJ, USA

Javier Sanchis Sáez Universitat Politecnica de Valencia, Valencia, Spain

Philipp Sauerteig Institute for Mathematics, Technische Universität Ilmenau, Ilmenau, Germany

Benjamin Seibold Department of Mathematics, Temple University, Philadelphia, PA, USA

Marwan A. Simaan University of Central Florida, Orlando, USA

Vijay K. Sood University of Ontario Institute of Technology, Oshawa, ON, Canada

Jonathan Sprinkle Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ, USA

Georgios E. Stavroulakis School of Production Engineering and Management, Technical University of Crete, Institute of Computational Mechanics and Optimization, Chania, Greece

Raphael Stern Department of Civil and Environmental Engineering, Institute for Software Integrated Systems, Vanderbilt University, Nashville, TN, USA

Georgios K. Tairidis School of Production Engineering and Management, Technical University of Crete, Institute of Computational Mechanics and Optimization, Chania, Greece

Daniel B. Work Department of Civil and Environmental Engineering, Institute for Software Integrated Systems, Vanderbilt University, Nashville, TN, USA

Karl Worthmann Institute for Mathematics, Technische Universität Ilmenau, Ilmenau, Germany

Ying Xu University of Central Florida, Orlando, USA

Chapter 1
Control Engineering from Classical to Intelligent Control Theory—An Overview

Maude Josée Blondin, Javier Sanchis Sáez and Panos M. Pardalos

Abstract Control engineering is the engineering discipline that refers to the use of automatic control. This discipline has been intensively enlarging over the past decades due to technological advances and technology affordability. Nowadays, almost all engineering activities exploit automatic control. Therefore, this chapter aims to provide the core knowledge concerning some of the most important features in control design and its methods. It covers basic information to introduce the readers to the other chapters of this volume. Fundamental system properties and specifications for control design such as robustness and stability are explained. In addition to a broad overview of modern control techniques with explicative examples and reference publications, the chapter focuses on four intelligent control techniques, which are fuzzy logic control, neural networks, metaheuristics control tuning, and multi-agent systems. Perspective and new trends of research are also exposed for each presented control technique as well as for control systems in general.

1.1 Introduction

Control engineering can be considered as an engineering discipline itself. This discipline refers to the use of automatic control to make systems or processes reach the desired behavior while operating under certain constraints. Control engineering has been intensively enlarging over the past decades due to the advancement of modern technologies and development of new systems, in particular intelligent systems. It has reached the point where control engineering has become an inherent part of

M. J. Blondin (✉) · P. M. Pardalos
University of Florida, Gainesville, Florida 32607, USA
e-mail: maude.blondin@ufl.edu

P. M. Pardalos
e-mail: pardalos@ise.ufl.edu

J. Sanchis Sáez
Universitat Politecnica de Valencia, Camí de Vera, s/n, 46022 Valencia, Spain
e-mail: jsanchis@isa.upv.es
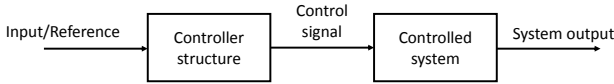
Fig. 1.1 Open-loop control system diagram [4]

almost all engineering activities covering a vast range of applications [1]. In practice, simple systems to more complex systems employ automatic control as, for example, a controlled on–off switch for temperature [2] to multi-input-multi-output (MIMO) industrial processes [3] that may be subject to several constraints.

There exist two categories that distinguished automatic control loop. The first is open-loop control—Fig. 1.1.

As shown in Fig. 1.1, open-loop control does not consider the system output in the control action. A toaster is one of the typical household items that applies the open-loop control concept. The system output, which is the bread's color, is excluded of the control action. Indeed, it is only a timer that controls the toaster [5]. The toasts pop up when the time is up, not when the desired bread's color is reached. Therefore, to assure that the desired color is reached, the user has to monitor the bread's color. It is only after a few trials the user will know how long he should set the timer to get "perfect" toasts. However, if another kind of bread is put in the toaster, the user will have to repeat the monitoring process. Monitoring the output to control a given system leads us to the concept of closed-loop control. In fact, monitoring the bread's color and adjust the timer can be considered a kind of closed-loop control known as "manual" control. Figure 1.2 presents the closed-loop control system diagram.

The control action depends on the system output feedback. A vehicle cruise control is a straightforward example of a closed-loop control system. The controller is the cruise control device. The reference is the desired speed set by the driver. The system output is the actual vehicle speed, which is continually monitored by a sensor. Both speeds are constantly compared. Their difference is the error, $e(s)$, that the controller tries to minimize by determining the adequate throttle position. Therefore, regardless of the road condition changes such as road slope and wind velocity, dynamic control actions will maintain the vehicle speed constant.
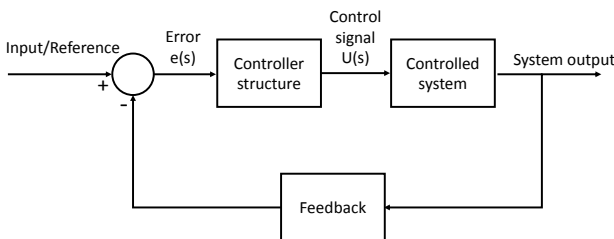
Fig. 1.2 Closed-loop control system diagram [4]

The best-known and most used feedback control structure is the proportional–integral–derivative (PID) controller. Indeed, a decade ago, PID controller was controlling more than 90% of practical control systems [6]. The non-interacting form in Laplace domain describes the PID algorithm as follows [7]:

$$PID(s) = \frac{U(s)}{e(s)} = K_p + \frac{K_i}{s} + K_d s \qquad (1.1)$$

The PID controller performance depends on the values of its proportional gain K_p, its integral gain K_i, as well as its derivative gain K_d. Only appropriate gains will provide the desired system responses. Consequently, PID controller parameters tuning is a crucial step in control design. Since there are only three tunable gains, one may think PID tuning is an easy task. In contrary, it is mostly challenging given that multiple criteria, which are often conflicting, must be satisfied. Indeed, fast dynamic system response is frequently at the expense of system robustness. Hence, the key is to achieve the best trade-off between system criteria. As results of seeking the best trade-off in control design requirements, a plethora of PID tuning methods have been proposed [8] and compared [9]. Classical methods and modern methods divide PID controller tuning methods. Table 1.1 presents an overview of the best-known PID tuning approaches.

Ziegler–Nichols (ZN) is the best-known heuristic tuning method. ZN determines the PID gains by following a set of heuristic rules. This method can also be applied to tune P and PI controllers. It is suitable for systems where the plant dynamics are not fully known [29]. Another well-known method is the pole placement, which is mainly applied to low-order systems. Specifically, this method positions the system closed-loop poles in the S-plane. The classical tuning methods presented in Table 1.1 are by design proposed for PID controllers. Even though they are usually suitable for proportional–integral (PI) controller, their applications are limited. Indeed, they are generally not appropriate for one-degree-of-freedom PID (1DOF-PID), two-degree-

Table 1.1 PID controller tuning methods

Classical methods	Ref.	Modern approaches	Ref.
Kappa-tau tuning	[7]	Fuzzy logic control	[19]
Ziegler–Nichols	[10]	Neural network theory	[20]
Pole placement	[11]	Machine learning	[21]
Gain and phase-based design	[12]	Multi-agent system	[22]
D-partitioning	[13]	**Metaheuristic tuning such as**:	[23]
Nyquist-based design	[14]	Particle swarm optimization	[24]
Cohen–Coon	[15]	Genetic algorithm	[25]
Internal model control	[16]	Ant colony optimization	[26]
Frequency-loop shaping	[17]	Fuzzy logic control + particle swarm optimization tuning	[27]
Cancelation-based	[18]	Fuzzy logic control + neural network	[28]

of-freedom PID (2DOF-PID), three-degree-of-freedom PID (3DOF-PID), fractional-order PID (FOPID), as well as for PID with a filter on the derivative term (PID/N). As other major drawbacks, they mainly rely on approximations of low-order models and hardly achieve high dynamic performance. Moreover, they can barely handle system constraints, which are ubiquitous and essential in practice [30, 31]. Indeed, any system has specific features and operating conditions that the control structure tuning must handle such as anti-windup mechanism. For these reasons, researchers and scientists have developed alternatives, so-called modern approaches, to cope better with system constraints while achieving high dynamic performance as well as meeting high stability and robustness criteria. For example, in [32], it has been demonstrated that a fuzzy logic controller-PID (FLC-PID) tuned by genetic algorithm (GA) provides better control compared to ZN-tuned PID controller for an inverted pendulum. In the same direction, in [33], particle swarm optimization (PSO)-tuned PID controller achieves better system performance than ZN-tuned and Cohen–Coon-tuned PID controller for an unmanned aerial vehicle camera position control. Another major advantage of modern methods is that some of them are suitable for controller structures unrelated to PID such as backstepping control structure [34]. Moreover, system optimality can be reached with modern methods. Therefore, they are of much interest in control design.

It is worth mentioning that no single PID tuning method performs best with all systems. This is also true for any other tuning methods applied to controller structures related or unrelated to PID controllers. Similarly, no single controller structure related or unrelated to PID is adequate to all systems. As a result, scientific community has been developing new controller structures as well as tuning techniques to cope better with system requirements.

In the light of this PID controller tuning review, challenges arise; which controller structure and tuning method should be selected. The selection of the control structure/algorithm along with its design depends on the system to be controlled and the closed-loop objectives. Therefore, this chapter aims to provide the reader with the core knowledge concerning some of the most important features in control design and to present the main modern control methods. It covers basic information to introduce the readers to the other chapters of this book. It provides details related to system properties and specifications for control design such as robustness and stability. In addition to a broad overview of modern control techniques, the chapter focuses on intelligent control techniques. Fuzzy logic control (FLC), neural network (ANN), metaheuristics tuning as well as multi-agent systems are detailed and situated into the literature. Moreover, perspective and new trends of research are also exposed for each presented technique.

Section 1.2 presents control system classifications, properties, and specifications. Section 1.3 presents the main modern control strategies with an emphasis on intelligent control methods.

1.2   Control System Classifications, Properties, and Specifications

This section presents the essential knowledge for control system design.

1.2.1   System Classification

There exist several ways to classify control systems. Consequently, system classification should be based on its purpose/use since there are more appropriate control methods for specific class. For instance, any system is either linear or nonlinear and time-invariant or time-varying. These classifications would help select the proper control design approach and tool analysis.

A system is linear if it obeys the superposition property defined by the homogeneity rule and the additivity rule. The homogeneity rule means that if a scalar b multiplies the input $X1$, the output $Y1$ will also be multiplied by b such that $F(bX1) = bF(X1)$. The additivity rule definition is if an input $X1$ produces the output $Y1$, and the input $X2$ produces the output $Y2$; therefore, the input $X1 + X2$ yields the output $Y1 + Y2$. In that direction, a system that does not comply with the superposition property is nonlinear. In a practical context, there is no completely linear system. All systems have some nonlinearities to a certain degree. However, it is more convenient to represent physical systems with linear equations since it simplifies their analysis and design. As a result, many systems are represented by linear equations. Sometimes linear equations may approximate nonlinear systems. This linearization process is possible only if the nonlinear system has an operation point around an equilibrium point with small variations around this point. Since the system linearization is valid only for a limited range of operations, a combination of linearized models can be used to define a larger operation range of the nonlinear system.

A time-invariant system as its name says does not change over time meaning that its characteristics stay the same. Thus, if a time shift is inserted in the input, the same is obtained in the output signal, regardless of when the input signal is applied [35].

Systems can also be classified according to their signal type either continuous-time signal or discrete-time signal. A signal is continuous if it exists for all time t. It is defined as $x(t)$. In the opposite, discrete signal, denoted with $x[n]$, is defined at distinct points of time. This latter is frequently the result of continuous-time signal sampling.

Another classification is single-input single-output (SISO) versus multiple-input multiple-output (MIMO) systems. SISO systems possess only one input and one output. In contrary, MIMO systems have several inputs and outputs. Further information on SISO vs. MIMO control design is presented in [36].

Proper classification of the system to control will help the control designer to select the appropriate control technique. Indeed, the control techniques applicable to a given system will be narrowed down by the system classification.

1.2.2 Transient Response—Dynamic Performance

In common practice, the following transient-response measurements to a step input are often used to characterize system dynamic performances [29]:

- Delay time, t_d: time required for the response to reach half of the final value for the first time [29].
- Rise time, t_r: time required for the response to rise from a certain percentage to another percentage of its final response value [29]. Usually, for overdamped systems, 10–90% rise time is employed and for underdamped second-order systems, 0–100% rise time is used.
- Peak time, t_p: time required for the response to reach the first peak of the overshoot [29].
- Maximum percent overshoot, M_p: this measure is computed as follows:

$$\text{Maximum percent overshoot} = \frac{\text{Maximum peak value} - \text{Final value}}{\text{Final value}} \times 100\%$$
(1.2)

- Settling time, t_s: time required for the response to reach and stay within a range around its final value [29]. The range is expressed in absolute percentage of the final value and is commonly either 2% or 5%.

The control designer could specify the values that these characteristics should take and then tune the controller structure with a selected technique. However, specifying the transient-response measurements might be unrealistic for systems with imprecisely known behavior. Therefore, these measurements are frequently used to compare tuning technique and controller structure performances. For instance, in [37], PID performances tuned by PSO, artificial bee colony optimization (ABC), and differential evolution (DE) are compared by using t_r, M_p, t_s and t_p values. For metaheuristics controller tuning, it is also current practice to design a performance criterion, also called cost function, objective function, or fitness function, using all or some of these measurements.

1.2.3 Stability

Stability is a key concept in designing a control system. At no time, a system should be or become unstable since unstable systems may have unexpected behavior. A system is absolutely stable if and only if any given bounded input produces a bounded output [35]. Such systems are called bounded input bounded output (BIBO). Let's consider the following system [35]:

$$y(t) = Hx(t)$$
(1.3)

This system is BIBO stable if its output meets the following condition:

$$|y(t)| \leqslant M_y < \infty \text{ for all } t \qquad (1.4)$$

whensoever the system input respects the condition:

$$|x(t)| \leqslant M_x < \infty \text{ for all } t \qquad (1.5)$$

where M_x and M_y are finite positive numbers. The closed-loop transfer function poles of absolutely stable systems are located in the left-half plane of the S-plane. There exist two other categories that classify system stability: (i) marginal stability and (ii) conditional stability. A system is marginally stable if the system produces an oscillatory output signal of constant amplitude and frequency for some bounded input. This kind of system has two closed-loop poles of its transfer function on the complex/imaginary axis. Conditionally stable systems refer to systems that are stable only under certain conditions.

Even though system stability can be checked by the poles' location, the following methods are often more straightforward to apply. Therefore, they are frequently used [4]:

- Routh–Hurwitz criterion: Algebraic method that determines the location of polynomial roots without solving the equation, i.e., without computing the zeros of the equation. The method specifies if either the roots are lying in the left or right half of the S-plane. This approach is suitable for SISO and MIMO systems as well as for single or multiple loop systems.
- Nyquist stability criteria: In addition to giving information concerning the absolute stability of a system as the Routh–Hurwitz criterion does, this method provides details on the frequency characteristics as well as on the degree of stability of a given system.
- Root locus: This technique allows the user to graph the location of the poles as some system parameters vary over a given range. The obtained graph is the diagram of loci. The system is stable only if the values of the selected parameters cause the poles' location lies in the left-half S-plane. Otherwise, the system is unstable.
- Bode diagram: Diagram called Bode plot includes two graphs: one indicating the magnitude of the transfer function of a system and the other presenting its phase.
- Lyapunov's stability criterion: This method indicates the stability of nonlinear and linear systems. A Lyapunov function is used to determine system stability properties.

1.2.4 Robustness

Such as stability, robustness is an important property in control design. Control is designed with a mathematical representation of the system. However, the system models are rarely exact. A modeled control system may have the desired behavior in

simulations, but offers the poorest performance in practice. Even with the most care in developing models, there is always some parameter variations and uncertainties in practice. This is one of the reasons any practical control system should have some robustness properties [38]. Robustness refers to the system's ability to cope with component variations while remaining stable as well as maintaining its performance. Robustness can also refer to the system being able to handle system disturbances. Depending on the control system application, disturbance rejection may also be critical. Robustness can be determined by using stability evaluation methods such as root locus. With the root locus graph, one can estimate the range of parameter values for which the system remains stable. In the same direction, one can use the phase and gain margins. These measurements provide information concerning how much additional gain and phase the system can take before it becomes unstable. This means that the system stability is more "robust" with high margins.

Frequently, dynamic performance, stability, and robustness are conflicting criteria. The key element in control design is achieving the best trade-off between these criteria for a given system [39]. Tuning methods as well as controller structures appropriateness depend on the system application and specifications. The following section provides good insight concerning the main modern control strategies and their applications as well as perspectives of research.

1.3 Modern Control Strategies

This section exposes the main modern control strategies for linear as well as nonlinear systems. Figure 1.3 presents a classification of modern control theory techniques with an emphasis on intelligent control.

Optimal control theory refers to techniques that are based on two fundamental ideas: (i) dynamic programming with the associated optimality principle and (ii) the maximum principle, namely, Pontryagin maximum principle [40] and Bellman approach, which only apply to deterministic problems [41]. For instance, in [42], dynamic programming for constrained optimal control problems is studied, where basic theoretical results and the description of the state-feedback optimal control law construction are provided. To limit typical phenomena of optimal control such as oscillations, concentrations, and discontinuities, a global unified methodology mostly based on the Hamilton–Jacobi–Bellman was proposed in [43]. This methodology lays foundation for the application of the Lasserre hierarchy to other optimal control problems as well as for calculus of variations. Research activities on optimal control have been very active over the past decades, which is substantiated by several surveys on optimal control and its major developments [44, 46–48]. A new research direction related to optimal control could be the codesign of optimal control techniques with other control approaches such as optimal control with backstepping design or with sliding mode control [44]. In the same direction, neural networks employed for optimal control synthesis is also a new tendency of research. For instance, an optimal adaptive neural network control scheme in finite horizon for nonlinear discrete-time

Modern control theory

Optimal control	Robust control	Intelligent control	Stochastic control	Adaptive control

Fuzzy control	Neural network	Metaheuristics tuning	Multi-agent system control
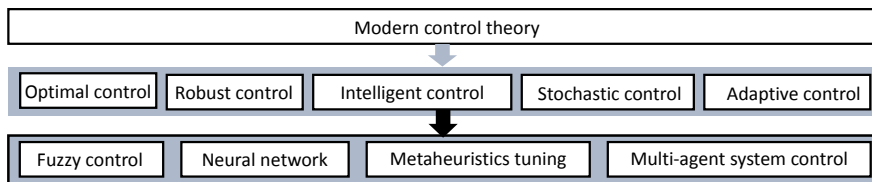
Fig. 1.3 Classification of control theory

systems is proposed in [45]. Fuzzy theory combined with optimal control was also recently developed. On the other hand, optimal control in cooperative control of nonlinear multi-agent systems deserves more investigation since a limited number of publications cover this topic.

Robust control is the control techniques specifically designed to deal with system uncertainties as well as disturbances [49]. Some examples of robust control approaches are sliding mode control and H-∞ loop shaping. The latter is one of the most powerful robust control techniques [50]. Research on robust control theory has been studied for more than 30 years. Research publications on the subject confirm that this topic is still of interest. In [51], a robust data-driven H-∞ feedback control is proposed for a wind turbine system. A robust internal model control based on sliding mode control approach for a servo motor is proposed in [52] to deal with system uncertainties and disturbances as well as to achieve high-performance motion control. Robot control is an area of research where robust control has been intensively studied. The publications [53, 54] are excellent surveys on this topic. However, robust control application is not exclusively for robots. Indeed, robust control can be applied also to a large range of applications such as photovoltaic systems [55], hydraulic excavators [56], and spacecraft [57].

Stochastic control class is mostly based on the characteristics of the system to be controlled rather than the techniques used. Indeed, stochastic control problems are presumed to have some random noise or disturbances in the model, which controllers have to consider. Therefore, stochastic control deals with controlling random systems in an optimal way such as controlling an airplane through turbulence [58].

Adaptive control refers to control systems with adjustable controllers, i.e., controller parameters are adapted/changed according to modifications in system dynamics along with system conditions [59]. For instance, in [60], a nonlinear adaptive controller is designed to handle all unknown and time-varying dynamics such as parameter uncertainties and nonlinearities as well as disturbances. This control system commands a permanent magnet synchronous generator-based wind turbine. Adaptive control tends to be combined with other control techniques. For example, in [61], a robust adaptive neural network control is proposed to control polymer electrolyte membrane fuel cells. Indeed, it is very frequent to have control designs that combine concepts of different classes of control theory. Codesigns are popular because it combines the strength of each control techniques. The last control category, intelligent control, has a strong tendency to be mixed with other control approaches. Intelli-

gent control, which includes fuzzy control, neural network, multi-agent systems, and controller tuning by metaheuristics, is detailed in the following subsections.

1.3.1 Intelligent Control

This section presents intelligent control methods and their applications. The intelligent control field has been expanding during the last decade due to the development of more complex systems [62]. As a system increases in complexity, its mathematical model also. In some cases, highly complex mathematical models may complicate its control design [63]. In a similar way, there are some systems that are difficult to model with differential or difference equations. Therefore, designing their control becomes very challenging with methods that rely on differential or difference equations. Moreover, as explained before, classical tuning methods, e.g., Ziegler–Nichols and Cohen–Coon, are not suitable for complex controller structures. Hence, scientific interest has grown tremendously in developing several methods based on intelligent control to address these limitations. Intelligent control refers to methodologies that use artificial intelligence techniques, which can be classified into fuzzy logic, neural networks, as well as metaheuristics tuning that include bio-inspired and evolutionary optimization algorithms. Cooperative control in multi-agent system context is also an active area of research that could be included in intelligent control methods. The subsequent subsections provide details, state of the arts, as well as future research direction for each of the abovementioned category.

1.3.1.1 Metaheuristics

Optimization plays a major and influential role in everyday life. Indeed, several fields such as engineering, economics, and computer science employ optimization. Optimization is applied to problems that need to achieve a certain optimally with respect to one or more objectives. Figure 1.4 presents a classification of optimization algorithms. Metaheuristics are characterized by a high degree of abstraction, which permits them to be easily adapted and implemented for a wide range of optimization problems. Specifically, they are designed to tackle complex and nonlinear problems wherein deterministic and heuristic optimization methods fail to yield the desired results [64]. In general, metaheuristics are relatively simple to implement and effectively bypass areas of local minima. For these reasons, over the last two decades, the development of new metaheuristics and their applications to controller tuning and design have gained in interest. This axis of research is still in constant innovation.

Metaheuristics can be categorized according to their foundation/source of inspiration [65]. The first category is based on natural laws of evolution. The optimization process starts with a population that evolves by combining the best individuals from one generation to another such as GA. Other metaheuristics are based on the principles of the laws of physics, e.g., SA algorithm. The third category is inspired by
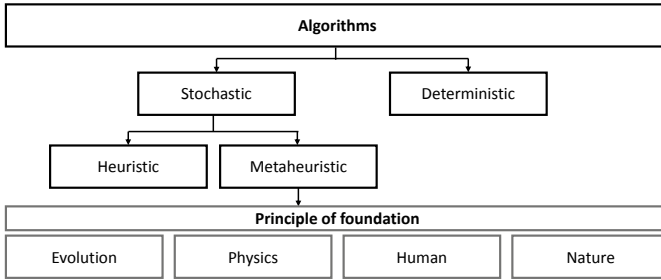
Fig. 1.4 Optimization algorithms classification

human behavior as, for instance, TLBO and SLC. The last category, which is nature-based, contains optimization methods that mimic the behavior of group of individuals such as insects and animals, e.g. PSO algorithm.

Regardless of the category, algorithm performance depends primarily on the balance between diversification process and intensification process. Diversification process deals with the exploration of new regions in the search space, either continuous or discrete, in order to generate solutions that are significantly different from the ones already created. The intensification process refers to the use of information already collected to intensify the search in areas of interest [66]. Reaching the equilibrium between these two processes is essential and achieved by an adequate parametrization of the algorithms, e.g., the number of individuals in a population such as in GA.

To optimize a problem by metaheuristics, the problem must be defined as follows:

$$
\begin{aligned}
\underset{x \Re^n}{\text{minimize}} \quad & f_i(\boldsymbol{x}) && (i = 1, 2, \ldots, M) \\
\text{subject to} \quad & \phi_j(\boldsymbol{x}) = 0 && (j = 1, 2, \ldots, J) \\
& \psi_k(\boldsymbol{x}) \leqslant 0 && (k = 1, 2, \ldots, K)
\end{aligned}
\tag{1.6}
$$

where $\boldsymbol{x}$ is the vector of n variables to optimize. A variable of $\boldsymbol{x}$ is represented by x_i. $f_i(\boldsymbol{x})$ refers to the functions to be minimized and are commonly called cost functions. $\phi_j(\boldsymbol{x})$ and $\psi_k(\boldsymbol{x})$ are the equality and inequality constraints, respectively. Figure 1.5 presents the design steps for controller tuning by metaheuristics [67, 68].

First, the designer must define the objectives to be minimized and constraints to be considered. When $M = 1$, the problem has only one cost function to minimize. For multi-objective problems, i.e., $M > 1$, there are two alternatives:

1. Cost functions are grouped into one function using weighting factors; or
2. Cost functions are simultaneously minimized and the designer chooses the solution that achieves the best trade-off between objectives by using, for instance, the Pareto front.

Cost function must be carefully designed. An inadequate cost function will lead the algorithm to an unsatisfactory solution. In other words, the algorithm may reach poor solutions due to an improper cost function and not because the algorithm is not suitable for the application.
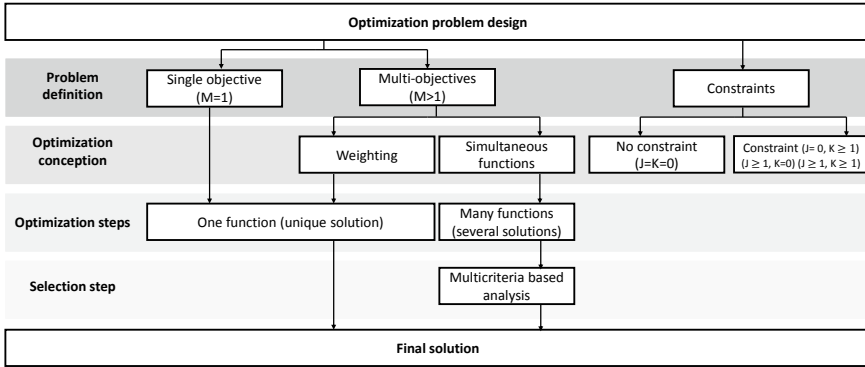
Fig. 1.5 Optimization problem design [31]

Table 1.2 Metaheuristics acronyms with reference works

Algorithm	Acronym	Ref.	Algorithm	Acronym	Ref.
Artificial bee colony optimization	ABC	[69]	Harmony search algorithm	HSA	[81]
Ant colony optimization	ACO	[70]	Imperialist competitive algorithm	ICA	[82]
Bat algorithm	BA	[71]	Mine blast algorithm	MBA	[83]
Bacterial foraging optimization algorithm	BFO	[72]	Multi-objective PSO	MOPSO	[84]
Cuckoo optimization algorithm	COA	[73]	Non-dominated sorting GA II	NSGA-II	[85]
Dragonfly algorithm	DA	[74]	Particle swarm optimization	PSO	[86]
Differential evolution	DE	[75]	Simulated annealing	SA	[87]
Firefly algorithm	FA	[76]	Teaching–learning-based optimization	TLBO	[88]
Fruit fly optimization algorithm	FOA	[77]	Tabu search algorithm	TSA	[89]
Flower pollination algorithm	FPA	[78]	Whale optimization algorithm	WOA	[65]
Genetic algorithm	GA	[79]	Two-lbests MOPSO	2LB-MOPSO	[90]
Grey wolf optimizer	GWO	[80]			

The problem is unconstrained only if $J = K = 0$. Otherwise, the problem is constrained by equality constraints if $J \geqslant 1$, $K = 0$, inequality constraints, if $J = 0$, $K \geqslant 1$ or both if $J \geqslant 1$ and $K \geqslant 1$.

There are several proposed metaheuristics in the literature. Chapter 2 of this volume presents in detail some of the main metaheuristics. Metaheuristics are tuning tools for control systems and not controller structures themselves.

Table 1.2 presents metaheuristic acronyms with reference works while Table 1.3 presents meta-analysis of original, modified, or combined metaheuristics applied to controller tuning. The term SOO refers to single objective optimization and MOO to multi-objective optimization.

This meta-analysis shows that metaheuristics are suitable for different controller structure tuning as well as for a wide range of applications. Combining algorithms together or modifying standard metaheuristics such as those presented in Table 1.3 may allow to reach better solutions than standard metaheuristics. Indeed, besides developing new algorithms, the trend research goes toward altering standard meta-heuristics or by blending standard or modified metaheuristics together. To choose the appropriate algorithm for a given system, the control designer finds an optimization algorithm that was applied to a similar system to the one to be controlled. Chapter 3 provides theoretical benchmark problems for controllers tuned by metaheuristics.

In the light of all these proposed algorithms presented in Table 1.3, more investigation on MOO should be performed since fewer publications are available on the subject compared to SOO. Moreover, MOO seems to yield better results. The research focus should also be on providing/developing theoretical information concerning metaheuristic components such as proof of convergence, parametrization algorithm framework along with algorithm limitations and specificities. Indeed, a limited number of works on these subjects are published. For instance, there are some parametrization guides available for GA and DE in [133, 134], respectively, but for most of the published algorithms, they are not any. Parametrization guides would help/ease the exploitation of the algorithm and their applications. Moreover, the focus should be on algorithm auto-tuning mechanism where the parametrization would evolve according to the complexity of the problem since most of the algorithms have constant parametrization throughout the optimization process. Scientific community should also agree on a common performance validation criteria in conjunction with benchmark problems and its setup to draw stronger conclusions concerning metaheuristics efficiency compared to one another [135].

1.3.1.2 Fuzzy Logic Control

Fuzzy logic control (FLC) is a promising option for systems that cannot be accurately modeled by mathematical equations such as nonlinear systems. FLC is also adequate for systems with important uncertainties or with contradicting criteria/conditions.

Computer programs make binary decisions, i.e., 0 or 1 that may be translated by true or false. In contrast, FLC has decision values between 0 and 1 [136]. For example, an air conditioner regulated by a thermostatic controller that has only two states: below or above the desired temperature translated as 0 or 1, respectively. There is no intermediate state or partially true answers possible. However, intermediate states exist with FLC. As a result, for a fuzzy air conditioner, the control designer would define grades between truly cold and truly hot temperature. For example, 0 could be assigned to 32° F, 0.25 to 50°, 0.50 to 75°, 0.75 to 95°, and 1 to 120°. The allocation of partially true values is strictly based on preferences without mathematical reasoning

Table 1.3 State of the art of metaheuristics applied to controller tuning

Ref.	Algorithm(s)	System	Controller(s)	SOO	MOO
[91]	GA, EP, PSO, ACO	Industrial processes	PID	–	–
[92]	GA-BF, PSO, GA-PSO, BF-GA	Automatic voltage regulator (AVR)	PID	––	✓
[93]	PSO, GA	Typical industrial models	PID	✓	–
[94]	GA	Automatic car parking mechanism	Neuro-fuzzy	✓	–
[95]	GA	Reverse osmosis plant	PID	✓	–
[96]	GA	Bidirectional inductive power transfer	PID	✓	–
[97]	GA	Power system	PI	–	✓
[98]	GA	Chemotherapy drugs	PID, I-PD	–	✓
[99]	GA	Gas turbine aeroengine	Fuzzy scheduling control	–	✓
[100]	2LB-MOPSO	Distillation column plant	PID	–	✓
[101]	NSGA-II	Induction motor	Predictive control	–	✓
[102]	NSGA-II	Synchronous generator excitation system	FOPID	–	✓
[103]	NSGA-II, MOPSO	Level control system	PI, PID, sliding mode control	–	✓
[104]	improved PSO	Hydraulic excavator	PID-based	✓	–
[105]	PSO and ABC	Benchmark problems	PID, FOPID	✓	–
[106]	Chaotic PSO	Hybrid power system	FO fuzzy PID	✓	–
[107]	PSO	Static synchronous compensator	PI	✓	–
[108]	BA, PSO, FPA, COA	Shell and tube heat exchanger	PI	–	✓
[109]	Lyapunov theory-based + PSO	Benchmark case studies	Fuzzy controller	–	✓
[110]	Dynamic GA-PSO	Power system	FO controller	✓	–
[111]	ABC, FPA	Permanent magnet synchronous motor	Linear quadratic regulator (LQR)	–	✓
[112]	PSO, ACO	Electric power-assisted steering	PID	✓	–
[113]	ACO	Mobile robot	FLC	✓	–
[114]	ACO-NM	AVR	PID, PID/N, 2DOF-PID	✓	–
[115]	ICA, BA	Plug-in hybrid electric vehicles	Model predictive control	✓	–

Table 1.3 (continued)

Ref.	Algorithm(s)	System	Controller(s)	SOO	MOO
[116]	Orthogonal SA	Various test plants	Fuzzy neural network for PID	✓	–
[117]	SA	AVR	PID	✓	–
[118]	SA	Nonlinear SISO and MIMO systems	Fuzzy systems	–	–
[119]	GWO	Nonlinear servo system	Fuzzy control system	✓	–
[120]	GWO, PSO	DC motor	PID	✓	–
[121]	WOA	Photovoltaic system	PI based	✓	–
[122]	DA	Hybrid energy distributed power system	3DOF-PID	✓	–
[123]	TLBO	Automatic generation control	FOPID	✓	–
[124]	TLBO	Dynamic voltage restorer	PI-based	✓	–
[125]	TLBO, BFA, FA, PSO	AVR	1DOF-PID, 2DOF-PID	✓	–
[126]	TLBO	AVR	PID	✓	–
[127]	Local unimodal sampling-TLBO	Multisource power system	Fuzzy PID-based	✓	–
[128]	FOA	Electronic throttle	FO fuzzy PID	✓	–
[129]	DE with spherical pruning	Industrial applications	FOPID	–	✓
[130]	ABC, GWO, COA, MBA, WOA, MSA	On-grid PV systems	PI, FOPI	✓	–
[131]	TSA	Benchmark process	PID-based	✓	–
[132]	HSA	Power system stabilizer	FLC	✓	–

[136]. The FLC makes decisions according to several preference rules defined by the control designer.

FLC has two major advantages: (i) the control logic is easy to read since it consists of *if-then* rules described with everyday vocabulary, and (ii) the controller can manage several inputs and outputs as well as resolve any conflicting criteria [136]. Figure 1.6 presents a way to classify FLC; conventional fuzzy control, adaptive fuzzy control and fuzzy control combined with other algorithms; more details concerning differences between categories are presented in [137, 138].

The conventional FLCs are Mamdani fuzzy control [139] and Takagi–Sugeno-Kang (TSK) fuzzy control [140]. The *if-then* rules are defined by the control designer and are fixed over time, which is different from adaptive FLC. The adaptive FLC
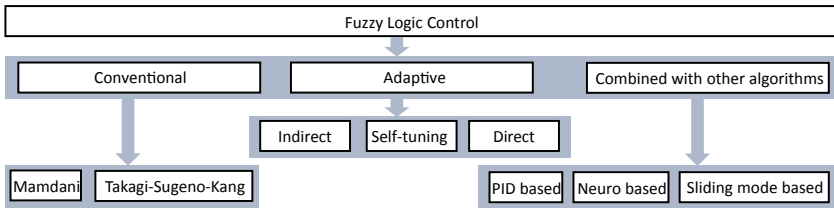
Fig. 1.6 FLC classification

can be divided into three categories: self-tuning fuzzy control and direct and indirect adaptive fuzzy control. The last category combines FLC with other algorithms.

Table 1.4 presents a short survey over the last decade on FLC for each category.

This review highlights only a part of fuzzy control and its application. However, it can be seen that FLC is employed in a wide range of applications from controlling robots to urban traffic networks. This subsection aimed to provide the basic knowledge and an overview of FLC. Chapter 4 of this volume covers FLC and its application to specific systems where fuzzification and defuzzification processes are explained.

Exhaustive surveys on FLC are available in the literature. Fuzzy control for industrial applications are presented in [159]. In [138], a survey on FLC-based control of marine surface vehicles and underwater vehicles is performed. Recent developments on the design and analysis of fuzzy model-based nonlinear networked control systems are presented in [160]. Advancements of analysis and design of model-based fuzzy control systems with an emphasis on stability and controller Takagi–Sugeno fuzzymodel-based design are surveyed in [161]. A state-of-the-art technique for identifying fuzzy models and designing model-based controllers is performed in [162]. Moreover, a survey of methods for FLC tuning is performed in [163].

Even though FLCs are suitable for control systems of diverse applications, there are still some challenges that merit study and further research to expand FLC applicability to practical systems. The challenge of choosing the most appropriate FLC for a given system arises from this plethora of proposed FLCs. Therefore, more emphasis should be dedicated to provide theoretical and mathematical frameworks on how to employ an already proposed FLC to a new application of the same type. In the same direction, more general fuzzy controllers suitable for a type of systems, for instance, should be designed instead of developing a specific FLC for a particular application. This would be beneficial to the scientific research community as well as for control designers/practitioners.

Table 1.4 Fuzzy logic controller

Category	Structure	Application	Ref.
Conventional	Takagi–Sugeno-type dynamic fuzzy model	Stochastic non-affine nonlinear systems	[141]
	Mamdani's FLC approach	Maximum power point tracking for photovoltaic system	[142]
	Mamdani-type fuzzy logic controller	Planar robot	[143]
	Zero-order Takagi–Sugeno–Kang-type FLC	Mobile robot navigation	[144]
Adaptive	Mamdani and Takagi–Sugeno-based direct adaptive FLC	DC motor	[145]
	Self-tuning fuzzy controller with a standard Takagi–Sugeno	Heat exchanger	[118]
	Direct adaptive fuzzy control	Third-order nonlinear system	[146]
	Fuzzy indirect adaptive controller	Duffing oscillator and inverted pendulum system.	[147]
	Fuzzy approximation-based indirect adaptive controller	Non-affine MIMO system and a two-link rigid robot manipulator	[148]
Combined	PID-like sliding mode fuzzy controller	Robot manipulator	[149]
	Fuzzy PID controller	Wind turbine pitch angle	[150]
	PID-type FLC	Multi-input multi-output active magnetic bearing system	[151]
	Sliding mode fuzzy controller	Boost converter	[152]
	Disturbance-observer-based PI-type fuzzy controller	Stochastic distribution systems	[153]
	Fuzzy-sliding and fuzzy-integral-sliding controller	Twin-rotor multi-input-multi-output system	[154]
	PID-like fuzzy	Quadrotor	[155]
	General type-II fuzzy logic-based controller tuned by modified backtracking search algorithm	Urban traffic network	[156]
	Cascade fuzzy controller based on differential evolution	Rotary inverted pendulum	[157]
	Fuzzy control learned through differential evolution	Orientation of a hexapod robot	[158]

1.3.1.3 Artificial Neural Networks

Artificial neural networks (ANNs) have brought a lot of attention to the research community. ANNs succeed where traditional control techniques are limited, mostly with nonlinear systems. Indeed, ANNs are effective in dealing with errors and uncertainties of modeling, system disturbances as well as unknown dynamics. Also, it has been demonstrated that ANNs provide great robustness in control systems [164]
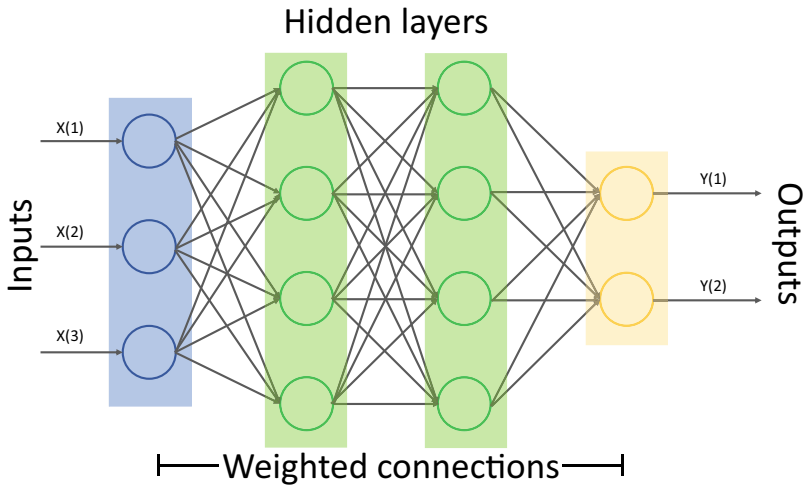
Fig. 1.7 Typical ANN—multilayer perceptron

such as in [165] where an ANN offered robust performance for autonomous vehicle parking control. For these reasons, control designers have employed ANNs in many control systems for diverse applications such as temperature control systems [166] and robotic arm control [167]. Indeed, several surveys for different applications have been published such as a survey on neural networks in process control [168], a comprehensive introduction and perspective of neural networks applications for power electronics and motor drives [169] and a review of neural networks for control systems [170]. Chapters 5 and 6 of this volume deal specifically with ANNs.

ANNs mimic the function of a brain. A typical ANN for control applications is the multilayer perceptron—Fig. 1.7 [164, 171]. It consists of several artificial neurons depicted as nodes and positioned in a series of layers. Edges define connections between neurons as synapses in the brain; they transmit signal/information from one neuron to another. Each edge has a certain weight. The majority of ANNs are fully connected weighted directed graphs. This means that each neuron in the hidden layer is connected with an associated weight to every neuron in the previous layer and the next layer. The first layer is the input to be processed by the ANN. The hidden layers convert/process the information received by the inputs. The last layer is the output containing the information needed by the system in which the ANN is used.

Any ANN must be trained. In other words, edge weights are tuned by using known data for the system under study. ANNs are well trained if they provide desired behavior with inputs that are outside of the training data. Backpropagation is the most popular training algorithm for multilayer perceptron [172]. Weights can also be adaptive meaning they are updated/adapted online while the networks are controlling the system. There exist other types of ANNs such as single-layer perceptron, radial basis networks, counterpropagation networks, recurrent neural networks, Hopfield network, modular neural networks, learning vector quantization, and adaptive reso-

Fig. 1.8 Soft computing framework for control design [175]
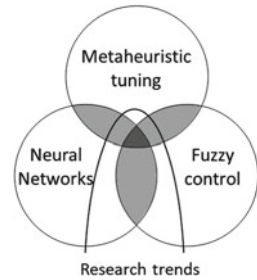
nance theory [164, 170]. In fact, there are more than 50 structures proposed in the literature [164]. The ANN application and objectives would help determine which ANN structure is the most appropriate for the control system.

As mentioned in the subsection on FLC, an area of research that has gained in interest lately is combining neural networks with fuzzy logic. These systems are called neuro-fuzzy systems (NFS) and aim to combine advantages of ANNs and FLC. As a result, they can provide better performance than ANN and FLC stand alone. For instance, in [173], Takagi–Sugeno dynamic neuro-fuzzy controller was developed to control an uncertain nonlinear system; the drawbacks of the individual implementation of the neural network and the fuzzy logic were overcome by the proposed hybrid structure. Chapter 4 of this volume presents in depth an application of neuro-fuzzy controllers. Along the same line, applying metaheuristics to neural networks has also attracted many researchers attention since improvement can be achieved. For example, in [174], a combination of an adaptive DE algorithm with backpropagation neural network was proposed to improve the forecasting accuracy of the neural network. In this context, Fig. 1.8 presents soft computing framework for intelligent control technique combination.

The combination of these three methods may facilitate the control design while providing better performance. For instance, in [94], GA tunes all parameters of a neuro-fuzzy logic controller (NFLC-GA). It has been demonstrated that the NFLC-GA provides better performance and is easier to design than a conventional fuzzy controller and a PID controller tuned by GA.

Combining two or three of the above intelligent control techniques is a research direction under constant innovation. Any new proposed hybrid controller brings some sort of improvement and novelty. However, there are more than a hundred metaheuristics proposed in literature [135] as well as a large number of fuzzy logic controllers and neural networks, which brings the number of possible combinations very high. Therefore, research and development should be on generalizing the combined-control methods to expand their application to other systems and ensure their practical implementation instead of developing a new hybrid controller for a particular application. Along the same line, theoretical application frameworks should be established to facilitate the employment of already proposed controllers and the ones under development.

1.3.1.4 Cooperative Control

Technological advances and technology affordability have enabled the deployment of multi-agent concept to many engineering activities from commercial to military applications. Consequently, cooperative control in a multi-agent system (MAS) context has been deeply studied over the past decades [176]. MAS is composed of several agents that work together to solve/accomplish a common task. An agent possesses the ability (i) to reason based on information, (ii) to receive feedback, and (iii) to act upon environment. Each agent makes decisions based on agent local information available to help the entire system achieve its goals. In other words, the control is decentralized [177]. In contrast, centralized systems have one decision-maker meaning that a central mechanism decides for the entire system.

Cooperative control can be split into two categories [183]. The first category is non-formation cooperative control problems. For instance, MAS concept has been applied in several power engineering applications [178] such as fault location systems for smart distribution grid [179] along with control and operation of microgrids [180]. The second category is formation control problems such as time-varying formation control for unmanned aerial vehicles [181].

Agents in cooperative control problems share information such as cost function values, relative positions, and velocities. One of the most studied problems in literature on cooperative control is the consensus problem. A group of agents must agree together to accomplish a particular task. The common schemes to multi-agent cooperative control are the following [182, 183]:

- Flocking—a team of agents interacts to align their speed, head the same direction, and keep relative distance constant to avoid collision.
- Rendez-vous problem—group of agents reaching a consensus on a position to meet.
- Synchronization of coupled oscillators—coupled oscillators synchronized their phases.
- Vehicle formations—group of vehicles achieving a specific formation and maintaining it while moving.

Several surveys on multi-agent control are available in the literature. A review of multi-agent system formation control with an emphasis on the sensing capability and interaction topology of agents is presented in [184]. A review of multi-agent systems for microgrid control is proposed in [180]. A review on control techniques of autonomous multi-agent quadrotor is available in [185]. In [186], a state-of-the-art survey manufacturing control system using multi-agent systems is performed. A review covering consensus and coordination of multi-agent systems can be found in [187]. A comparative review of multi-agent cooperative control consensus is presented in [176]. Multi-agent coordination consensus problems are surveyed in [183]. From these surveys, mutual challenges in multi-agent perspective have been identified regardless of the schemes of cooperation control. System constraints, disturbances, time delay, communication lost, communication noise, model uncertainties as well as inconsistency in time delay, for instances, are some challenges that should

be more investigated [183]. In this direction, global stability properties specifically for distance-based control such as flocking outside triangular formation deserves more investigation [180]. Moreover, a large number of publications present theoretical research of consensus problems with simulation results. However, more experimental results would help increase commercial and military implementation of the proposed techniques. Therefore, research publication should include practice validations.

Related to MAS, Chapter 7 of this volume deals with cooperative control and distributed optimization while Chapter 8 presents distributed optimization schemes embedded in model predictive control implementations.

1.4 Conclusion

This chapter has provided an overview from traditional to modern control techniques with an emphasis on intelligent control methods. Core knowledge on system classification, properties, and specifications was also exposed. Modern control techniques were briefly explained with informative references. In particular, four categories of intelligent control methods were further explained. Short literature reviews were also presented. The perspective of research and the new trends were also concisely presented. This overview does not pretend to be exhaustive, but we expect that this chapter provided fundamental knowledge and explanation to help design control systems and introduce the readers to the other chapters of this volume. It is important to mention that the modern control techniques are not exclusive to the classification presented in his chapter. Indeed, Chapters 9, 10, 11, 12, 13 and 14 of this volume present relevant control techniques that overlap and/or exceed the presented classification.

From this control engineering review, general perspective and research direction could be drawn. Reducing the gap between theoretical research and practical implementation would enhance research value. One approach could be establishing theoretical and mathematical frameworks of the control techniques and mathematically specify/define the systems the control methods could be applied to. In the same direction, while developing new control methods to solve a control problem, scientific community should also focus on keeping the methods suitable to other systems. In summary, these suggestions would increase application possibilities to other systems in addition to commercial or military implementations.

References

1. Kozák, S.: State-of-the-art in control engineering. Journal of Electrical Systems and Information Technology **1(1)**, 1-9 (2014)
2. Golob, M. Tovornik, B. and Donlagic, D.: Comparison of the self-tuning on-off controller with the conventional switching controllers. First IEEE Conference on Control Applications, pp. 962-963 (1992)
3. Chai, T., Joe Qin, S. and Wang, H.: Optimal operational control for complex industrial processes. Annual Reviews in Control, **38(1)**, 81–92 (2014)
4. Kuo, B.C.: Automatic control systems. Prentice-Hall Englewood Cliffs, N.J (1975)
5. Ulusoy, M.: Understanding Control Systems. IN : MathWorks - Videos and Webinar https://www.mathworks.com/videos/understanding-control-systems-part-1-open-loop-control-systems-123419.html https://www.mathworks.com/videos/understanding-control-systems-part-2-feedback-control-systems-123501.html
6. Li, Y., Ang, K.H., Chong, G.C.: Patents, software, and hardware for PID control: an overview and analysis of the current art. IEEE Control Systems Magazine **26**(1), 42–54 (2006)
7. Aström, K.J, Hägglund, T.: PID controllers: theory, design, and tuning. In: Research Triangle Park, NC: Instrument society of America, 1-354, (2012)
8. Cominos, P., and Munro, N.: PID controllers: recent tuning methods and design to specification, In: IEEE Proceedings-Control Theory and Applications, **149(1)**, pp. 46-53 (2002)
9. Ribeiro, J. M. S. Santos, M.F., Carmo, M.J., Silva, M.F.: Comparison of PID controller tuning methods: analytical/classical techniques versus optimization algorithms. In: IEEE 18th international Carpathian control conference (ICCC), pp. 533-538. (2017)
10. Ziegler, J.G., and Nichols, N.B. : Optimum settings for automatic controllers. trans. ASME **64(11)**, (1942)
11. Aström, K.J., Hägglund, T., Hang, C.C., Ho, W.O.: Automatic tuning and adaptation for PID controllers - a survey. Control Engineering Practice **1**(4), 699–714 (1993)
12. Ho, W.K., Xu, W.: PID tuning for unstable processes based on gain and phase-margin specifications. IEE Proceedings-Control Theory and Applications **145**(5), 392–396 (1998)
13. Jinggong, L., Yali, X., Donghai, L.: Calculation of PI controller stable region based on D-partition method. ICCAS 2010, Gyeonggi-do, pp. 2185-2189, (2010)
14. Chen, D., Seborg, D.E.: Design of decentralized PI control systems based on Nyquist stability analysis. Journal of Process Control **13**(1), 27–39 (2003)
15. Cohen, G.: Theoretical consideration of retarded control. Trans. Asme. **75**, 827–34 (1953)
16. Rivera, D.E., Morari, M., Skogestad, S.: Internal model control: PID controller design. Industrial & engineering chemistry process design and development **25**(1), 252–265 (1986)
17. Grassi, E., et al.: Integrated system identification and PID controller tuning by frequency loop-shaping. IEEE Transactions on Control Systems Technology **9(2)**, 285–294 (2001)
18. Isaksson, A.J., Graebe, S.F.: Analytical PID parameter expressions for higher order systems. Automatica **35**(6), 1121–1130 (1999)
19. Visioli, A.: Tuning of PID controllers with fuzzy logic. IEE Proceedings-Control Theory and Applications **148**(1), 1–8 (2001)
20. Fang, M.-C., Zhuo, Y.-Z., Lee, Z.Y.: The application of the self-tuning neural network PID controller on the ship roll reduction in random waves. Ocean Engineering **37**(7), 529–538 (2010)
21. Zulu, A.: Towards explicit PID control tuning using machine learning. In: 2017 IEEE AFRICON, pp. 430–433 (2017)
22. Yuan, S., Zhao, C., Guo, L.: Decentralized PID control of multi-agent systems with nonlinear uncertain dynamics. In: 2017 36th Chinese Control Conference (CCC), pp. 8857–8862 (2017)
23. Ghosal, S., Darbar, R., Neogi, B., Das, A., Tibarewala, D.N.: Application of swarm intelligence computation techniques in PID controller tuning: a review. In: Proceedings of the International Conference on Information Systems Design and Intelligent Applications. Springer, Berlin, pp. 195–208 (2012)

24. Gaing, Z.L.: A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Trans. Energy Convers. **19**(2), 384–391 (2004)
25. Zhang, J., Zhuang, J., Du, H.: Self-organizing genetic algorithm based tuning of PID controllers. Inf. Sci. **179**(7), 1007–1018 (2009)
26. Hsiao, Y.-T., Chuang, C.-L., Chien, C.C.: Ant colony optimization for designing of PID controllers. In: 2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No. 04CH37508) (2004)
27. Yang, M., Wang, X.: Fuzzy PID controller using adaptive weighted PSO for permanent magnet synchronous motor drives. In: 2009 Second International Conference on Intelligent Computation Technology and Automation, pp. 736–739 (2009)
28. Kelly, D.J., Burton, P.D., Rahman, M.A.: The application of a neural-fuzzy logic controller to process control. In: Proceedings of the First International Joint Conference of The North American Fuzzy Information Processing Society Biannual Conference, pp. 235–236 (1994)
29. Ogata, K.: Modern Control Engineering, 4th edn. Prentice Hall PTR, USA (2014)
30. Padula, F., Visioli, A.: Optimal tuning rules for proportional-integral-derivative and fractional-order proportional-integral-derivative controllers for integral and unstable processes. IET Control. Theory Appl. **6**(6), 776–8 (2012)
31. Reynoso-Meza, G., Sanchis, J., Blasco, X., Martínez, M.: Algoritmos Evolutivos y su empleo en el ajuste de controladores del tipo PID: Estado actually perspectivas. Revista Iberoamericana de Automática e Informática Industrial RIAI **10**(3), 251–68 (2013)
32. Mukherjee, S., Pandey, S., Mukhopadhyay, S., Hui, N.B.: Digital pendulum system: genetic fuzzy-based online tuning of PID controller. In: 2014 IEEE 8th International Conference on Intelligent Systems and Control, pp. 23–28 (2014)
33. Rajesh, R.J., Ananda, C.M.: PSO tuned PID controller for controlling camera position in UAV using 2-axis gimbal. In: IEEE International Conference on Power and Advanced Control Engineering (ICPACE), pp. 128–133 (2015)
34. Yueneng, Y.A., Ye, Y.A.: Backstepping sliding mode control for uncertain strict-feedback nonlinear systems using neural-network-based adaptive gain scheduling. J. Syst. Eng. Electron. **29**(3), 580–6 (2018)
35. Haykin, S., Van Veen, B.: Signals and Systems. Wiley, New York (2005)
36. Goodwin, G.C., Graebe, S.F., Salgado, M.E.: Control System Design. Prentice Hall, NJ (2001)
37. Gozde, H., Cengiz Taplamacioglu, M., Kocaarslan, I.: Comparative performance analysis of artificial Bee colony algorithm in automatic generation control for interconnected reheat thermal power system. Int. J. Electr. Power Energy Syst. **42**(1), 16–178 (2012)
38. Stengel, R.F.: Intelligent failure-tolerant control. IEEE Control Syst. Mag. **11**(4), 14–23 (1991)
39. Garpinger, O., Hägglund, T., Aström, K.J.: Performance and robustness trade-offs in PID control. J. Process Control **24**(5), 568–577 (2014)
40. Soliman, S.A., Mantawy, A.A.: Modern Optimization Techniques with Applications in Electric Power Systems. Springer Science & Business Media, Berlin (2011)
41. Todorov, E.: Optimal Control Theory. Bayesian Brain: Probabilistic Approaches to Neural Coding, pp. 269–98 (2006)
42. Borrelli, F., Baotić, M., Bemporad, A., Morari, M.: Dynamic programming for constrained optimal control of discrete-time linear hybrid systems. Automatica **41**(10), 1709–1721 (2005)
43. Henrion, D., Kružžík, M., Tillmann, W.: Optimal control problems with oscillations, concentrations and discontinuities. Automatica **103**, 159–165 (2019)
44. Zhang, Y., Li, S., Liao, L.: Near-optimal control of nonlinear dynamical systems: A brief survey. Ann. Rev. Control (2019)
45. Zhao, Q., Hao, X., Sarangapani, J.: Neural network-based finite-horizon optimal control of uncertain affine nonlinear discrete-time systems. IEEE Trans. Neural Netw. Learn. Syst. **26**(3), 486–499 (2015)
46. Agarwal, R.P., Baleanu, D., Nieto, J.J., Torres, D.F., Zhou, Y.: A survey on fuzzy fractional differential and optimal control nonlocal evolution equations. J. Comput. Appl. Math. **339**, 3–29 (2018)

47. Rao, A.V.: A survey of numerical methods for optimal control. Adv. Astronaut. Sci. **135**(1), 497–528 (2009)
48. Minchala-Avila, L.I., Garza-Castañón, L.E., Vargas-Martínez, A., Zhang, Y.: A review of optimal control techniques applied to the energy management and control of microgrids. Procedia Comput. Sci. **52**, 780–787 (2015)
49. Bhattacharyya, S.P., Keel, L.H.: Robust Control: The Parametric Approach. Advances in Control Education, pp. 49–52. Pergamon (1994)
50. Vinnicombe, G.: Uncertainty and Feedback: H-∞ Loop-shaping and the [nu]-gap Metric. World Scientific (2001)
51. Kim, Y.M.: Robust data driven H-infinity control for wind turbine. J. Frankl. Inst. **353**(13), 3104–3117 (2016)
52. Li, P., Zhu, G.: Robust internal model control of servo motor based on sliding mode control approach. ISA Trans. (2019)
53. Abdallah, C., Dawson, D.M., Dorato, P., Jamshidi, M.: Survey of robust control for rigid robots. IEEE Control Syst. Mag. **11**(2), 24–30 (1991)
54. Sage, H.G., De Mathelin, M.F., Ostertag, E.: Robust control of robot manipulators: a survey. Int. J. Control **72**(16), 1498–1522 (1999)
55. Deveci, O., Kasnakoğlu, C.: MIMO nonlinear modeling and robust control of standalone dc photovoltaic systems. Int. J. Hydrog. Energy **42**(28), 18064–18080 (2017)
56. Kim, S., Park, J., Kang, S., Kim, P.Y., Kim, H.J.: A robust control approach for hydraulic excavators using μ-synthesis. Int. J. Control. Autom. Syst. **16**(4), 1615–1628 (2018)
57. Malekzadeh, M.: Robust control of spacecraft: application to an actuated simulator. Int. J. Control. Autom. Syst. **16**(2), 896–903 (2018)
58. Harris, N.: Stochastic Control (2011)
59. Aström, K.J., Wittenmark, B.: Adaptive Control. Courier Corporation (2013)
60. Chen, J., Yao, W., Zhang, C.K., Ren, Y., Jiang, L.: Design of robust MPPT controller for grid-connected PMSG-Based wind turbine via perturbation observation based nonlinear adaptive control. Renew. Energy **134**, 478–495 (2019)
61. Abbaspour, A., Khalilnejad, A., Chen, Z.: Robust adaptive neural network control for PEM fuel cell. Int. J. Hydrog. Energy **41**(44), 20385–20395 (2016)
62. Tzafestas, G., Spyros and others: Methods and Applications of Intelligent Control (1997)
63. Siddique, N.: Intelligent Control: A Hybrid Approach Based on Fuzzy Logic, Neural Networks and Genetic Algorithms. Springer, Berlin (2013)
64. Xiong, N., Molina, D., Ortiz, M.L., Herrera, F.: A walk into metaheuristics for engineering optimization: principles, methods and recent trends. Int. J. Comput. Intell. Syst. **8**(4), 606–636 (2015)
65. Mirjalili, S., Lewis, A.: The whale optimization algorithm. Adv. Eng.Softw. **1**(95), 51–67 (2016)
66. Di Tollo, G., Lardeux, F., Maturana, J., Saubion, F.: An experimental study of adaptive control for evolutionary algorithms. Appl. Soft Comput. **35**, 359–72 (2015)
67. Reynoso-Meza, G., Blasco, X., Sanchis, J., Martínez, M.: Controller tuning using evolutionary multi-objective optimisation: current trends and applications. Control Eng. Practice **28**, 58–73 (2014)
68. Yang, X.S.: Engineering Optimization: An Introduction with Metaheuristic Applications. Wiley, New York (2010)
69. Karaboga, D., Basturk, B.: Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. In: International Fuzzy Systems Association World Congress, pp. 789–798. Springer, Berlin (2007)
70. Dorigo, M., Birattari, M.: Ant Colony Optimization. Springer, Berlin (2010)
71. Yang, X.S.: A new metaheuristic bat-inspired algorithm. Nature Inspired Cooperative Strategies For Optimization, pp. 65–74. Springer, Berlin (2010)
72. Das, S., Biswas, A., Dasgupta, S., Abraham, A.: Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications. Found. Comput. Intell. **3**, 23–55 (2009)
73. Rajabioun, R.: Cuckoo optimization algorithm. Appl. Soft Comput. **11**(8), 5508–18 (2011)

74. Mirjalili, S.: Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Comput. Appl. **27**(4), 1053–73 (2017)
75. Price, K., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Springer Science & Business Media, Berlin (2006)
76. Fister, I., Fister Jr., I., Yang, X.S., Brest, J.: A comprehensive review of firefly algorithms. Swarm Evol. Comput. **13**, 34–46 (2013)
77. Pan, W.T.: A new fruit fly optimization algorithm: taking the financial distress model as an example. Knowl.-Based Syst. **26**, 69–74 (2012)
78. Yang, X.S.: Flower pollination algorithm for global optimization. In: International Conference on Unconventional Computing and Natural Computation, pp. 240–249. Springer, Berlin (2012)
79. Davis, L.: Handbook of Genetic Algorithms (1991)
80. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014)
81. Geem, Z.W., Kim, J.H., Loganathan, G.V.: A new heuristic optimization algorithm: harmony search. Simulation **76**(2), 60–68 (2001)
82. Atashpaz-Gargari, E., Lucas, C.: Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: IEEE Congress on Evolutionary Computation, pp. 4661–4667 (2007)
83. Sadollah, A., Bahreininejad, A., Eskandar, H., Hamdi, M.: Mine blast algorithm: a new population based algorithm for solving constrained engineering optimization problems. Appl. Soft Comput. **13**(5), 2592–612 (2013)
84. Coello, C.C., Lechuga, M.S.: MOPSO: A proposal for multiple objective particle swarm optimization. In: Proceedings of the 2002 Congress on Evolutionary Computation, vol. 2, pp. 1051–1056 (2002)
85. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: International Conference on Parallel Problem Solving from Nature, pp. 849–858. Springer, Berlin (2000)
86. Kennedy, J.: Particle swarm optimization. In: Encyclopedia of Machine Learning, pp. 760–766 (2010)
87. Van Laarhoven, P.J., Aarts, E.H.: Simulated Annealing. Simulated Annealing: Theory and Applications, pp. 7–15. Springer, Berlin (1987)
88. Rao, R.V., Savsani, V.J., Vakharia, D.P.: Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. Comput.-Aided Des. **43**(3), 303–15 (2011)
89. Glover, F.: Tabu search-part I. ORSA J. Comput. **1**(3), 190–206 (1989)
90. Zhao, S.Z., Suganthan, P.N.: Two-lbests based multi-objective particle swarm optimizer. Eng. Optim. **43**(1), 1–7 (2011)
91. Nagaraj, B., Murugananth, N.: A comparative study of PID controller tuning using GA, EP, PSO and ACO. In: IEEE International Conference On Communication Control And Computing Technologies, pp. 305–313 (2010)
92. Kim, D.H.: Hybrid GA-BF based intelligent PID controller tuning for AVR system. Appl. Soft Comput. **11**(1), 11–22 (2011)
93. Ou, C., Lin, W.: Comparison between PSO and GA for parameters optimization of PID controller. In: IEEE International Conference on Mechatronics and Automation, pp. 2471–2475 (2006)
94. Seng, T.L., Khalid, M.B., Yusof, R.: Tuning of a neuro-fuzzy controller by genetic algorithm. IEEE Trans. Syst. Man Cybern. Part B (Cybernetics) **29**(2), 226–236 (1999)
95. Kim, J.S., Kim, J.H., Park, J.M., Park, S.M., Choe, W.Y., Heo, H.: Auto tuning PID controller based on improved genetic algorithm for reverse osmosis plant. World Acad. Sci. Eng. Technol. **47**(2), 384–9 (2008)
96. Ayala, H.V., Dos Santos Coelho, L.: Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator. Expert Syst. Appl. **39**(10), 8968–74 (2012)

97. Panda, S.: Multi-objective evolutionary algorithm for SSSC-based controller design. Electr. Power Syst. Res. **79**(6), 937–44 (2009)

98. Algoul, S., Alam, M.S., Hossain, M.A., Majumder, M.A.: Multi-objective optimal chemotherapy control model for cancer treatment. Med. Biol. Eng. Comput. **49**(1), 51–65 (2009)

99. Chipperfield, A.J., Bica, B., Fleming, P.J.: Fuzzy scheduling control of a gas turbine aeroengine: a multiobjective approach. IEEE Trans. Ind. Electron. **49**(3), 536–48 (2002)

100. Zhao, S.Z., Iruthayarajan, M.W., Baskar, S., Suganthan, P.N.: Multi-objective robust PID controller tuning using two lbests multi-objective particle swarm optimization. Inform. Sci. **181**(16), 3323–35 (2011)

101. Guazzelli, P.R., Pereira, W.C., Oliveira, C.M., Castro, A.G., Aguiar, M.L.: weighting factors optimization of predictive torque control of induction motor by multi-objective genetic algorithm. IEEE Trans. Power Electron. (2018)

102. Kumar, L., Narang, D.: Tuning of fractional order PI D controllers using evolutionary optimization for pid tuned synchronous generator excitation system. IFAC-PapersOnLine. **51**(4), 859–64 (2018)

103. Laware, A.R., Talange, D.B., Bandal, V.S.: Evolutionary optimization of sliding mode controller for level control system. ISA Trans. **83**, 199–213 (2018)

104. Ye, Y., Yin, C.B., Gong, Y., Zhou, J.J.: Position control of nonlinear hydraulic system using an improved PSO based PID controller. Mech. Syst. Signal Process. **83**, 241–59 (2017)

105. Bingul, Z., Karahan, O.: Comparison of PID and FOPID controllers tuned by PSO and ABC algorithms for unstable and integrating systems with time delay. Optim. Control Appl. Methods **39**(4), 1431–50 (2018)

106. Pan, I., Das, S.: Fractional order fuzzy control of hybrid power system with renewable generation using chaotic PSO. ISA Trans. **62**, 19–29 (2016)

107. Liu, C., Hsu, Y.: Design of a self-tuning PI controller for a STATCOM using particle swarm optimization. IEEE Trans. Ind. Electron. **57**(2), 702–715 (2010)

108. Castelo Damasceno, N., Gabriel Filho, O.: PI controller optimization for a heat exchanger through metaheuristic bat algorithm, particle swarm optimization, flower pollination algorithm and cuckoo search algorithm. IEEE Lat. Am. Trans. **15**(9), 1801–1807 (2017)

109. Das Sharma, K., Chatterjee, A. Rakshit, A.: A hybrid approach for design of stable adaptive fuzzy controllers employing lyapunov theory and particle swarm optimization. IEEE Trans. Fuzzy Syst. **17**(2), 329–342 (2009)

110. Kuttomparambil Abdulkhader, H., Jacob, J. Mathew, A.T.: Fractional-order lead-lag compensator-based multi-band power system stabiliser design using a hybrid dynamic GA-PSO algorithm. IET IET Gener., Transm. Distrib. **12**(13), 3248–3260 (2018)

111. Tarczewski, T., Grzesiak, L.M.: An application of novel nature-inspired optimization algorithms to auto-tuning state feedback speed controller for PMSM. IEEE Trans. Ind. Appl. **54**(3), 2913–2925 (2018)

112. Hanifah, R.A., Toha, S.F., Ahmad, S., Hassan, M.K.: Swarm-intelligence tuned current reduction for power-assisted steering control in electric vehicles. IEEE Trans. Ind. Electron. **65**(9), 7202–7210 (2018)

113. Castillo, O., Neyoy, H., Soria, J., Melin, P., Valdez, F.: A new approach for dynamic fuzzy logic parameter tuning in ant colony optimization and its application in fuzzy control of a mobile robot. Appl. Soft Comput. **28**, 150–9 (2015)

114. Blondin, M.J., Sanchis, J., Sicard, P., Herrero, J.M.: New optimal controller tuning method for an AVR system using a simplified Ant colony optimization with a new constrained Nelder-Mead algorithm. Appl. Soft Comput. **62**, 216–29 (2018)

115. Elsisi, M., Soliman, M., Aboelela, M.A.S., Mansour, W.: Model predictive control of plug-in hybrid electric vehicles for frequency regulation in a smart grid. IET Gener. Transm. Distrib. **11**(16), 3974–3983 (2017)

116. Ho, S., Shu, L.-S., Ho, S.Y.: Optimizing fuzzy neural networks for tuning PID controllers using an orthogonal simulated annealing algorithm OSA. IEEE Trans. Fuzzy Syst. **14**(3), 421–434 (2006)

117. Abido, M.A.: Simulated annealing based approach to PSS and FACTS based stabilizer tuning. Int. J. Electr. Power Energy Syst. **22**(4), 247–58 (2000)
118. Jain, R., Sivakumaran, N., Radhakrishnan, T.K.: Design of self tuning fuzzy controllers for nonlinear systems. Expert Syst. Appl. **38**(4), 4466–76 (2011)
119. Precup, R.E., David, R.C., Petriu, E.M.: Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity. IEEE Trans. Ind. Electr. **64**(1), 527–34 (2017)
120. Madadi, A., Motlagh, M.M.: Optimal control of DC motor using grey wolf optimizer algorithm. Tech. J. Eng. Appl. Sci. **4**(4), 373–9 (2014)
121. Hasanien, H.M.: Performance improvement of photovoltaic power systems using an optimal control strategy based on whale optimization algorithm. Electr. Power Syst. Res. **157**, 168–76 (2018)
122. Guha, D., Roy, P.K., Banerjee, S.: Optimal tuning of 3 degree-of-freedom proportional-integral-derivative controller for hybrid distributed power system using dragonfly algorithm. Comput. Electr. Eng. **72**, 137–53 (2018)
123. Gorripotu, T.S., Samalla, H., Rao, C.J., Azar, A.T., Pelusi, D.: TLBO algorithm optimized fractional-order PID controller for AGC of interconnected power system. In: Soft Computing in Data Analytics, pp. 847–855. Springer, Berlin (2019)
124. Khalghani, M.R., Khooban, M.H.: A novel self-tuning control method based on regulated bi-objective emotional learning controller's structure with TLBO algorithm to control DVR compensator. Appl. Soft Comput. **24**, 912–22 (2014)
125. Rajinikanth, V., Satapathy, S.C.: Design of controller for automatic voltage regulator using teaching learning based optimization. Procedia Technol. **21**, 295–302 (2015)
126. Chatterjee, S., Mukherjee, V.: PID controller for automatic voltage regulator using teaching-learning based optimization technique. Int. J. Electr. Power Energy Syst. **77**, 418–29 (2016)
127. Sahu, B.K., Pati, T.K., Nayak, J.R., Panda, S., Kar, S.K.: A novel hybrid LUS-TLBO optimized fuzzy-PID controller for load frequency control of multi-source power system. Int. J. Electr. Power Energy Syst. **74**, 58–69 (2016)
128. Sheng, W., Bao, Y.: Fruit fly optimization algorithm based fractional order fuzzy-PID controller for electronic throttle. Nonlinear Dyn. **73**(1–2), 611–9 (2013)
129. Sánchez, H.S., Padula, F., Visioli, A., Vilanova, R.: Tuning rules for robust FOPID controllers based on multi-objective optimization with FOPDT models. ISA Trans. **66**, 344–61 (2017)
130. Ramadan, H.S.: Optimal fractional order PI control applicability for enhanced dynamic behavior of on-grid solar PV systems. Int. J. Hydrog. Energy **42**(7), 4017–31 (2017)
131. Bagis, A.: Tabu search algorithm based PID controller tuning for desired system specifications. J. Frankl. Inst. **348**(10), 2795–812 (2011)
132. Sambariya, D.K., Prasad, R.: Optimal tuning of fuzzy logic power system stabilizer using harmony search algorithm. Int. J. Fuzzy Syst. **17**(3), 457–70 (2015)
133. Blasco Ferragud, F.X.: Control predictivo basado en modelos mediante técnica de optimización heurística: Aplicación a procesos no lineales y multivariables. Ph.D. Thesis 1999 (in Spanish) Editorial UPV. ISBN 84-699-5429-6
134. Pedersen, M.E.: Good Parameters for Differential Evolution. Magnus Erik Hvass Pedersen (2010)
135. Hussain, K., Salleh, M.N., Cheng, S., Shi, Y.: Metaheuristic research: a comprehensive survey. Artif. Intell. Rev. 1–43 (2018)
136. Jantzen, J.: Foundations of Fuzzy Control: A Practical Approach. Wiley, New York (2013)
137. Martinez, O.A., Cardona, M.: State of the art and future trends on unmanned aerial vehicle. In: 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE) (2018)
138. Xiang, X., Yu, C., Lapierre, L., Zhang, J., Zhang, Q.: Survey on fuzzy-logic-based guidance and control of marine surface vehicles and underwater vehicles. Int. J. Fuzzy Syst. **20**(2), 572–86 (2018)
139. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. Int. J. Man Mach. Stud. **7**(1), 1–13 (1975)

140. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Syst. Man Cybern. **15**(1), 116–132 (1985)
141. Gao, Q., Feng, G., Wang, Y., Qiu, J.: Universal fuzzy models and universal fuzzy controllers for stochastic nonaffine nonlinear systems. IEEE Trans. Fuzzy Syst. **21**(2), 328–341 (2013)
142. Messai, A., Mellit, A., Guessoum, A., Kalogirou, S.A.: Maximum power point tracking using a GA optimized fuzzy logic controller and its FPGA implementation. Solar Energy **85**(2), 265–77 (2011)
143. Bingül, Z., Karahan, O.: A fuzzy logic controller tuned with PSO for 2 DOF robot trajectory control. Expert Syst. Appl. **38**(1), 1017–31 (2011)
144. Juang, C.F., Chang, Y.C.: Evolutionary-group-based particle-swarm-optimized fuzzy controller with application to mobile-robot navigation in unknown environments. IEEE Trans. Fuzzy Syst. **19**(2), 379–92 (2011)
145. Zaki, A.M., El-Bardini, M., Soliman, F.A., Sharaf, M.M.: Embedded two level direct adaptive fuzzy controller for DC motor speed control. Ain Shams Eng. J. **9**(1), 65–75 (2018)
146. Chen, B., Liu, X., Liu, K., Shi, P., Lin, C.: Direct adaptive fuzzy control for nonlinear systems with time-varying delays. Inform. Sci. **180**(5), 776–92 (2010)
147. Boulkroune, A., Bounar, N., Farza, M.: Indirect adaptive fuzzy control scheme based on observer for nonlinear systems: a novel SPR-filter approach. Neurocomputing **135**, 378–87 (2014)
148. Boulkroune, A., M'Saad, M., Farza, M.: Fuzzy approximation-based indirect adaptive controller for multi-input multi-output non-affine systems with unknown control direction. IET Control Theory Appl. **6**(17), 2619–29 (2012)
149. Piltan, F., Sulaiman, N., Gavahian, A., Soltani, S., Roosta, S.: Design mathematical tunable gain PID-like sliding mode fuzzy controller with minimum rule base. Int. J. Robot. Autom. **2**(3), 146–56 (2011)
150. Civelek, Z., Lüy, M., Cam, E., Barisci, N.: Control of pitch angle of wind turbine by fuzzy PID controller. Intell. Autom. Soft Comput. **22**(3), 463–71 (2016)
151. Noshadi, A., Shi, J., Lee, W.S., Shi, P., Kalam, A.: Optimal PID-type fuzzy logic controller for a multi-input multi-output active magnetic bearing system. Neural Comput. Appl. **27**(7), 2031–46 (2016)
152. Guo, L., Hung, J.Y., Nelms, R.M.: Comparative evaluation of sliding mode fuzzy controller and PID controller for a boost converter. Electr. Power Syst. Res. **81**(1), 99–106 (2011)
153. Yi, Y., Zheng, W.X., Sun, C., Guo. L.: DOB fuzzy controller design for non-gaussian stochastic distribution systems using two-step fuzzy identification. IEEE Trans. Fuzzy Syst. **24**(2), 401–418 (2016)
154. C. Tao, Taur, J., Chang, Y., Chang, C.: A novel fuzzy-sliding and fuzzy-integral-sliding controller for the twin-rotor multi-input-multi-output system. IEEE Trans. Fuzzy Syst. **18**(5), 893–905 (2010)
155. Santos, M., Lopez, V., Morata, F.: Intelligent fuzzy controller of a quadrotor. In: IEEE International Conference on Intelligent Systems and Knowledge Engineering, pp. 141–146 (2010)
156. Khooban, M.H., Vafamand, N., Liaghat, A., Dragicevic, T.: An optimal general type-2 fuzzy controller for urban traffic network. ISA Trans. **66**, 335–43 (2017)
157. Oh, S.K., Kim, W.D., Pedrycz, W.: Design of optimized cascade fuzzy controller based on differential evolution: simulation studies and practical insights. Eng. Appl. Artif. Intell. **25**(3), 520–32 (2012)
158. Juang, C.F., Chen, Y.H., Jhan, Y.H.: Wall-following control of a hexapod robot using a data-driven fuzzy controller learned through differential evolution. IEEE Trans. Ind. Electron. **62**(1), 611–9 (2015)
159. Precup, R.E., Hellendoorn, H.: A survey on industrial applications of fuzzy control. Comput. Ind. **62**(3), 213–26 (2011)
160. Qiu, J., Gao, H., Ding, S.X.: Recent advances on fuzzy-model-based nonlinear networked control systems: a survey. IEEE Trans. Ind. Electron. **63**(2), 1207–17 (2016)
161. Feng, G.: A survey on analysis and design of model-based fuzzy control systems. IEEE Trans. Fuzzy Syst. **14**(5), 676–97 (2006)

162. Sala, A., Guerra, T.M., Babuška, R.: Perspectives of fuzzy systems and control. Fuzzy Sets Syst. **156**(3), 432–44 (2005)
163. Strietzel, R.: Tuning of fuzzy controllers-an overview. IFAC Proc. Vol. **33**(25), 313–318 (2000)
164. Marugán, A.P., Márquez, F.P., Perez, J.M., Ruiz-Hernández, D.: A survey of artificial neural network in wind energy systems. Appl. Energy **228**, 1822–36 (2012)
165. Heinen, M.R., Osório, F.S., Heinen, F.J., Kelber, C.: Seva3d: Using artificial neural networks to autonomous vehicle parking control. In: IEEE International Joint Conference on Neural Network Proceedings, pp. 4704–4711 (2006)
166. Khalid, M., Omatu, S.: A neural network controller for a temperature control system. IEEE Control Syst. Mag. **12**(3), 58–64 (1992)
167. Ligutan, D.D., Abad, A.C., Dadios, E.P.: Adaptive robotic arm control using artificial neural network. In: IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), pp. 1–6 (2018)
168. Thibault, J., Grandjean, B.P.: Neural networks in process control-a survey. In: Advanced Control of Chemical Processes, pp. 251–260. Pergamon (1992)
169. Bose, B.K.: Neural network applications in power electronics and motor drives-an introduction and perspective. IEEE Trans. Ind. Electron. **54**(1), 14–33 (2007)
170. Hunt, K.J., Sbarbaro, D., Zbikowski, R., Gawthrop, P.J.: Neural networks for control systems-a survey. Automatica **28**(6), 1083–112 (1992)
171. Hagan, M.T., Demuth, H.B., Jesús, O.D.: An introduction to the use of neural networks in control systems. Int. J. Robust Nonlinear Control.: IFAC-Affil. J. **12**(11), 959–85 (2002)
172. Balakrishnan, S.N., Weil, R.D.: Neurocontrol: a literature survey. Math. Comput. Model. **23**(1–2), 101–17 (1996)
173. Cervantes, J., Yu, W., Salazar, S., Chairez, I.: Takagi-Sugeno dynamic neuro-fuzzy controller of uncertain nonlinear systems. IEEE Trans. Fuzzy Syst. **25**(6), 1601–15 (2017)
174. Wang, L., Zeng, Y., Chen, T.: Back propagation neural network with adaptive differential evolution algorithm for time series forecasting. Expert Syst. Appl. **42**(2), 855–63 (2015)
175. Cordón, O., Gomide, F., Herrera, F., Hoffmann, F., Magdalena, L.: Ten years of genetic fuzzy systems: current framework and new trends. Fuzzy Sets Syst. **141**(1), 5–31 (2004)
176. Gulzar, M.M., Rizvi, S.T., Javed, M.Y., Munir, U., Asif, H.: Multi-agent cooperative control consensus: a comparative review. Electronics **7**(2), 1–22 (2018)
177. Weidlich, A., Vogt, H., Krauss, W., Spiess, P., Jawurek, M., Johns, M., Karnouskos, S.: Decentralized intelligence in energy efficient power systems. In: Handbook of Networks in Power Systems I, pp. 467–486. Springer, Berlin (2012)
178. McArthur, S.D., Davidson, E.M., Catterson, V.M., Dimeas, A.L., Hatziargyriou, N.D., Ponci, F., Funabashi, T.: Multi-agent systems for power engineering applications-Part I: concepts, approaches, and technical challenges. IEEE Trans. Power Syst. **22**(4), 1743–52 (2007)
179. Pang, Q., Gao, H., Minjiang, X.: Multi-agent based fault location algorithm for smart distribution grid. In: IET Conference Proceedings (2010)
180. Kantamneni, A., Brown, L.E., Parker, G., Weaver, W.W.: Survey of multi-agent systems for microgrid control. Eng. Appl. Artif. Intell. **45**, 192–203 (2015)
181. Dong, X., Yu, B., Shi, Z., Zhong, Y.: Time-varying formation control for unmanned aerial vehicles: theories and applications. IEEE Trans. Control Syst. Technol. **23**(1), 340–8 (2015)
182. Olfati-Saber, R., Fax, J.A., Murray, R.M.: Consensus and cooperation in networked multi-agent systems. Proc IEEE **95**(1), 215–33 (2007)
183. Ren, W., Beard, R.W., Atkins, E.M.: A survey of consensus problems in multi-agent coordination. In: IEEE American Control Conference, pp. 1859–1864 (2005)
184. Oh, K.K., Park, M.C., Ahn, H.S.: A survey of multi-agent formation control. Automatica **53**, 424–40 (2015)
185. Nathan, P.T., Almurib, H.A., Kumar, T.N.: A review of autonomous multi-agent quad-rotor control techniques and applications. In: 4th International Conference on Mechatronics, pp. 1-7 (2011)

186. Leitão, P.: Agent-based distributed manufacturing control: a state-of-the-art survey. Eng. Appl. Artif. Intell. **22**(7), 979–91 (2009)
187. Qin, J., Ma, Q., Shi, Y., Wang, L.: Recent advances in consensus of multi-agent systems: a brief survey. IEEE Trans. Ind. Electron. **64**(6), 4972–83 (2017)

Chapter 2
Main Metaheuristics Used
for the Optimization of the Control
of the Complex Systems

Pierre Borne and Amira Gharbi

Abstract Many optimization problems are usually NP-hard problems which prevent the implementation of exact solution methodologies. This is the reason why engineers prefer to use metaheuristics which are able to produce good solutions in a reasonable computation time. The metaheuristic approaches can be separated into two classes: the local search techniques and the global ones. Among the local search techniques, the taboo search and the simulated annealing are the most known. A possible acceleration of the convergence can be obtained by using tunneling algorithms. Concerning the global methods, the Genetic or Evolution Algorithms (GA), Ant Colony Optimization (ACO), and the Particle Swarm Optimization (PSO) are the most known.

2.1 Introduction

Many optimization problems are usually NP-hard problems which prevent the implementation of exact solution methodologies. This is the reason why engineers prefer to use metaheuristics which are able to produce good solutions in a reasonable computation time. The metaheuristic approaches can be separated into two classes: the local search techniques and the global ones. Among the local search techniques, the taboo search and the simulated annealing are the most known. A possible acceleration of the convergence can be obtained by using tunneling algorithms. Concerning

P. Borne · A. Gharbi (✉)

Centre de recherche en informatique, signal et automatique deLille.

Ecole Centrale de Lille, Cité scientifique, 48-59651 Villeneuve d'Ascq Cédex, France

e-mail: merkarim@gmail.com

P. Borne
e-mail: pierre.borne@ec-lille.fr

the global methods, the Genetic or Evolution Algorithms (GA), Ant Colony Optimization (ACO), and the Particle Swarm Optimization (PSO) are the most known.

Multi-objective optimization can be approached using Ordered Weighted Averaging (OWA) approach, Choquet integral, and Pareto optimality.

In this chapter we only present the various metaheuristics with only very small and easy to check examples. Among various important applications we can cite as examples [1–7]. This chapter is a combination of many papers [8–10] and it is organized as follows.

2.2   Local Methods

2.2.1   *Taboo Search*

The origin of taboo search traces to the greedy descent algorithm.

Assume that the goal is to minimize the f criterion. Then search for minimal point is performed over the vicinity $V(x_i)$ where x_i is the current solutions and a better solution x_j is obtained if $f(x_j) < f(x_i)$, the approach is easy to implement but there is a risk of rapidly stopping the search on a local minimum.

The taboo search method was proposed to avoid this problem [11–17], by using memory of the previously obtained solutions in order to permit to escape to a local optimum. It is an iterative local search procedure which enables to move from a solution to another solution in its neighborhood until a stopping criterion is satisfied. In practice, the main taboo search approach consists of determining, starting from a solution, the best solution in its immediate neighborhood with interdiction to go to one of the N previously obtained solutions. Let us denote $N(x)$ the list of the N solutions that have been visited in the recent past. With this method, we avoid having a cyclic evolution; it can appear that during some time we can have a degradation of the solution, but it enables us to get out of a local optimum and enlarge the search space.

The definition and the size of the taboo list are important parameters of the search.

If the list is too large, then the search is restricted to small area, but the risk to miss the global optimum becomes important; on the contrary, if the list is too small, it is very likely that the search is slowed down by a loop. The common approach is to have a constant length taboo list. In this case, the most recently visited solution enters the taboo list while the oldest solution is removed from the list as soon as the list has reached its predefined maximum length. A better approach consists of defining extreme of the taboo list, $N_{min} \leq N \leq N_{max}$ to make it evolve according to the following rules:

- If the currently generated solution improves the criterion, the taboo list is decreased; the two oldest solutions are removed from the list while the most recent solution enters into the list according to the predetermined constraints on N.

- If the currently generated solution does not improve the criterion, then the length is increased; this solution enters the list without suppression of the oldest solution (while always $N_{min} \leq N \leq N_{max}$).

In order to increase the diversification of the search, the usual approaches are as follows:

- Sudden change of focus where there are solutions not yet visited,
- Reinitializations of the search selected at random,
- Exclusion of the most visited solutions, and
- Penalties applied to the solutions near the current solution.

The stop test can be a maximum number of iterations or a maximum of solutions which do not improve the criterion performance.

This algorithm is very easy to implement for a problem with a small number of variables.

2.2.2 Simulated Annealing

Simulated annealing [18–23] is a generic probabilistic algorithm developed to solve local optimization problems for a function defined in a large search space. Simulated annealing has obtained excellent results in various complex problems. It is inspired by the physical thermic annealing. At each step of calculation of this algorithm, the current solution is replaced by a near one chosen with a probability that depends on the variation of a fitness function (called the energy function by analogy with the physical process, via a parameter T (called the temperature) which gradually and regularly decreases during the process.

In this approach, the solution changes almost randomly for large values of T and tends globally to obtain the minimum of the energy function as T approaches zero. The random evolution enables motions in which the energy can sometimes increase which avoids falling and being trapped in a local minimum which can appear with a usual downhill method as the gradient method. This algorithm can be presented as follows: let us denote by s, T, and e, respectively, the current state, temperature, and energy and s_n and e_n, respectively, the new state and energy.

The process is initialized with $s := s_0$ and $e := e_0$ which correspond to the initial state s_0 of energy e_0 at time k_0. While the stopping condition is not satisfied (time $k < k_m$ and energy $e > e_m$, with k_m the maximum number of allowed iteration and e_m minimum value of expected energy), we choose some state in the neighborhood and compute its energy.

Example of a simulation annealing algorithm:

Initialization

$$s = s_0, e = E(s), k = 0, T = T_0 , \tag{2.1}$$

while $k < k_m$ and $e > e_m$

$$s_{k+1} = neighbor(s_k) \tag{2.2}$$

$$e_{k+1} = E(s_{k+1}), e_k = e_{k+1} - e_k \tag{2.3}$$

If random

$$[0.1] < \exp\left(-\frac{e_k}{T_k}\right) \tag{2.4}$$

then

$$s_k := s_{k+1}, e_k := e_{k+1}, T_{k+1} := T_k, k := k+1 \text{ return while} \tag{2.5}$$

After stabilization, decrease T and return to s

We always save the best solution that will be the final solution given by the simulated annealing algorithm.

As for the previous algorithm, this one is very easy to implement for a problem with a small number of variables.

2.2.3 Tunneling Algorithms

Tunneling algorithms [24–28] (Fig. 2.1) enable to escape local optima. The idea is the following: each time a local optimum is reached the algorithm bore, a tunnel toward a new valley of the objective function $f(x)$

Originally, tunneling approaches had been defined for problems with continuous variables and were adapted to combinatorial problems later.

Two main strategies have been proposed, the stochastic tunneling and the tunneling with penalties functions.

The stochastic tunneling was initially defined to escape from local minima when implementing the simulated annealing algorithms at low temperature. The idea was to

Fig. 2.1 Tunneling algorithms
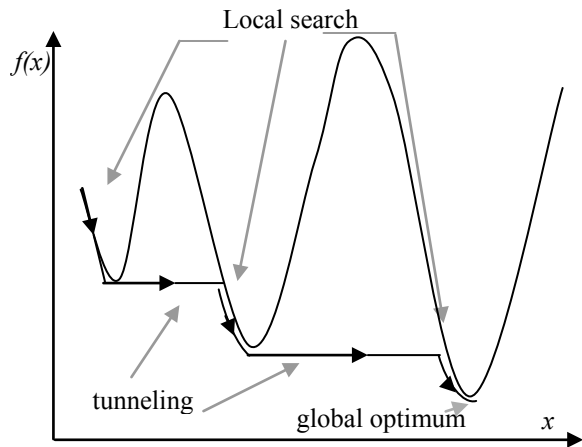
circumvent the slow dynamics of ill-shaped energy functions by applying a nonlinear transformation to the objective function.

The tunneling with penalty function modifies the value of local optima by adding penalty values in order to facilitate the algorithm to escape from local optima.

As an example, if $f(x)$ is the fitness function, we can choose the new fitness function

$$f_m(x) : f_m(x) = 1 - \exp(-\gamma(f(x) - f(x^*)))), \tag{2.6}$$

where x^* corresponds to the best known solution, and $\gamma > 0$ is a parameter defining the deforming degree of search space (unit by default).

This algorithm enables to accelerate the research of the solution but the implementation is usually limited to a problem with a maximum of three variables.

2.2.4   GRASP Methods

The greedy randomized adaptive search procedure consists of starting from a number of initial solutions to perform the local optimization and to apply each time the greedy descent algorithm. These initial solutions are generated at random.

2.3   Global Methods

2.3.1   Genetic Algorithms and Evolutionary Algorithms

The genetic and evolutionary algorithms are based on the mechanisms of genetics and the natural selection which exist in biology. Globally life is evolving toward its best to an optimum. Therefore, it is interesting to simulate the evolution manner of living entities in order to solve some optimization problems. For this reason, heuristic methods described in this chapter are referred to as evolutionary approaches or strategy of evolution.

The following vocabulary is used:

- Chromosome: alpha-numeric encoding of the solution;
- Gene: functional block of chromosome that encodes a specific property;
- Allele: a characteristic which is encoded by a gene;
- Population: a collection of chromosomes;
- Generation: a structure of population at a specific instant;
- Parents: chromosomes involved in the reproduction of the new individuals in a population;
- Children or offspring: chromosomes resulting from reproduction;
- Selection: mechanism of population renewal across generations;

Fig. 2.2 Two points
crossover

Parent1

Parent2

Child1

Child2
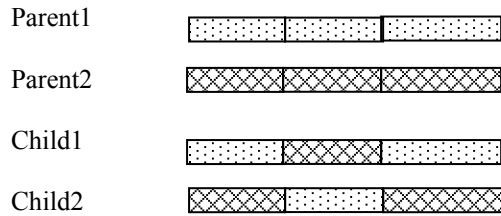
- Viability: the probability of a chromosome to survive and reproduce;
- Fitness: a measure of the quality of an individual (chromosome);
- Crossover: exchange of genes between chromosomes;
- Mutation: random change of some genes in a chromosome;

2.3.1.1 Crossover

The crossover corresponds to an exchange of genes usually between two individuals (parents) of the population. The two parents are selected according to their fitness via a probability defined by the roulette wheel and the crossover points can be decided with a stochastic approach or using special rules. The resulting off springs (children) incorporate genes from both parents as illustrated in Fig. 2.2.

The crossing area is determined by two parameters: the pivot which indicates the position of the gene in the chromosome and the length which precise the number of positions involved in the crossover. The crossover between chromosomes can be generalized, for example by considering multiple pivots and lengths randomly chosen.

Another generalization consists in using a masked crossover. The mask is a virtual binary chromosome whose unit values indicate the position to be exchanged in both parents.

2.3.1.2 Mutation

Mutation is a random change of one or more alleles in a chromosome. It aims to prevent the pauperization of the population's genetic heritage. In reasonably small proportion, the mutants are benefit to preserve certain diversity in the population, which avoids its rapid degeneration to a local optimum. Usually, the mutation is performed by using a mask that indicates the start of the genes that can be mutated.

The mask is either selected at random or according to chromosome specific internal structure.

Another possibility of mutation corresponds to the permutation of two sets of genes in the chromosome.

The inversion can also be implemented; it produces a change in the concatenation of the genes in certain areas of a chromosome, so that the new gene sequence is inverted with respect to the initial ones.

2.3.1.3   Selection

Selection for the Reproduction

−Elitist selection

The N_B best individuals in the sense of fitness can be chosen for reproduction. But this approach can reduce the diversity of the population and leads to a local optimum.

−Selection with the roulette wheel

If we search a maximum of the fitness and that the individual x_i has the fitness $f(x_i), i = (1, \ldots, N)$,
then individual x_i is selected for the reproduction with probability

$$p_i, \quad p(x_i) = \frac{f(x_i)}{\sum\limits_{j=1}^{N} f(x_j)} \tag{2.7}$$

−Selection by ranking

It is possible to assign each individual a rank, which generally is determined by its fitness.
The population can be defined by $P = \{x_1, x_2, \ldots, x_N\}$
with

$$f(x_1) \geq f(x_2) \geq \cdots \geq f(x_N) \tag{2.8}$$

Then the rank of the individual x_r being equal to r.
   The probability that this individual can be chosen for the reproduction is p_r

$$p_r = \left(1 - \frac{r}{\upsilon}\right)^p \tag{2.9}$$

with p positive constant and υ number of the best considered individuals. If the whole population is considered $\upsilon = N$

−Selection by tournament

We choose at random a number $N_r < N$ of individuals in the population and the best individual of this group is chosen. We repeat the operation till we have the wanted number of individuals for the reproduction.

Selection for Survival

−Generational selection

The inheritors are preferred regardless of their fitness. For the crossover, the children will replace the parents and for the mutation the mutant will replace the initial individual. This choice is intended to maintain the diversity of the population. However, the major risk is to remove the current or the global optimum.

- Elitist selection

Only the N best individuals among parents, children, and mutants are chosen to define the new population. With this approach, the search for the optimum can be slow.

- Generational elitist selection

The technique appears generally to be the best. The best of the reproducers and inheritors group are selected for the new population.

2.3.1.4 General Algorithm

The algorithm is initialized by a population that can be determined by another approach or whose individuals are randomly generated. Starting from this initial population, new generations are created from which the fitness of every individual is evaluated.

The implementation of the genetic algorithm can be summarized as follows, Fig. 2.3:

1. Create an initial population.
2. Evaluate the fitness of each individual of this population.
3. While the terminating condition is not satisfied, repeat:

   - Select best ranking individuals to reproduce.
   - Breed new generation through crossover and mutation to create offspring.
   - Evaluate the fitness of the new individuals.
   - Replace worst-ranked individuals of the population by the new ones.

In practice, the algorithm needs to be adapted to the specificities of the studied problem and in particular crossover and mutation are to be defined in order to create viable individuals satisfying all the conditions needed for the specific problem [29–32].

The more important is to choose the good chromosome for the encoding of the solution. The mutations which correspond to the random change of some characteristics of individual ensure to maintain a sufficient diversity in the population and avoid converging prematurely toward local optima rather than the global optimum of the problem.

This algorithm was initially defined for the optimization of discrete systems and has been adapted for continuous ones.

Fig. 2.3 Genetic algorithms
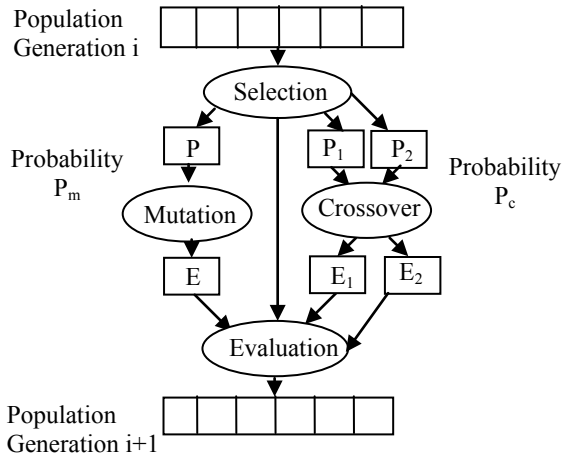
2.3.2   Ant Colony Optimization (ACO)

2.3.2.1   Principle

The ant colony optimization is inspired by ability and the organization of real ant colony using external chemical pheromone trails acting as a means of communication [33–36].

Ants are quickly able to find the shortest path from the nest to a food source.

The explanation is as follows:

- Ants go in randomly chosen directions, each one laying a pheromone trail on their paths;
- As soon as one of them found the food, it returns to the nest deposing pheromone again;
- The ants prefer to follow the paths with the highest concentration of pheromone;
- The new ants who leave the nest tend to paths with the greatest concentration of pheromone.

In fact pheromone trails slowly evaporate, reducing its attractive strength, so the more time it takes for an ant to achieve its trip to the food, the more time have the pheromone to evaporate.

The evaporation is necessary in order to avoid the premature convergence to a locally optimal solution.

Once an ant has found a short path from the nest to a food source, other ants prefer to follow that path which involves a positive feedback so that finally all the ants will follow this single path.

Artificial ant colonies have additional properties:

- Each ant has its own memory to keep in mind the traveled paths and to evaluate the current solution for which the pheromone trace has to be strengthened at the end of its journey.
- The pheromone trace can be updated each time an ant has traveled a full itinerary, by ascertaining the quality of the found solution.
- The ants move in a graph along arcs with labels set by the intensity of pheromone traces left by the others ants.
- The ants have the capacity to facilitate the exploration of paths that have not been considered yet.

2.3.2.2 Phonomone Trail Update

Let us denote $\tau_{ij}(t)$ the pheromone trail on the arc ij of a path at time t and

$$\Delta\tau_{ij}(t+1) = \tau_{ij}(t+1) - \tau_{ij}(t) \tag{2.10}$$

We have

$$\tau_{ij}(t) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t+1) \tag{2.11}$$

where ρ is the coefficient of evaporation of the pheromone.

On Line Update

In such a case, we have

$$\Delta\tau_{ij}(t+1) = \tau_f s_{k+1} = neighbor(s_k) \tag{2.12}$$

where $\tau_f > 0$ is a constant

Adaptative Delayed Update

During the current exploration, let L^f be the length or cost of the path for the ant f and L^b be the length of the ant having traveled a path of minimum length or cost v

The pheromone deposit of ant f on the arc ij is

$$\Delta\tau_{ij}^f(t+1) = \delta\frac{L^f}{L^b} \tag{2.13}$$

With $\delta \in (0, 1)$

Update by Ant Ranking

The ants are sorted in descending order according to the performance of traveled paths, with the best ants on first position

Denote $r(f)$ the rank of ant f and $\Delta\tau_{ij}^{r(f)}(t+1)$ the pheromone deposit of ant f who traveled on the arc ij and n_b the number of the best ants

It comes

$$\tau_{ij+1}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{r=1}^{n_b}(n_b - r + 1)\Delta\tau_{ij}^r(t+1) \qquad (2.14)$$

Update Through Elitist Strategy

Let L^b the length or cost of the best ant b

A supplementary amount of pheromone

$$\Delta\tau_{ij+1}(t+1) = \frac{Q}{L^b} \qquad (2.15)$$

is added on all the arcs ij visited by the ant b with Q a constant which depends on the natural capacity of the colony.

Initially, defined for discrete problems, this algorithm has been generalized for continuous one's.

2.3.3 Particle Swarm Optimization (PSO)

2.3.3.1 Principle

Particle swarm optimization [37–43] is a population based stochastic optimization technique. It is founded on the notion of cooperation between agents (the particles) that can be seen as animals with limited intellectual capacities: small memory and small intelligence. The exchange of information between them permits nevertheless that globally they succeed to solve difficult problems as it appears with bees, fishes, or birds. It appears that social sharing of information among individuals in competition offers an evolutionary advantage. In the particle swarm optimization algorithm, Fig. 2.4 particles move in multidimensional space and are characterized by a position and a velocity. They have two essential reasoning capabilities: the memory of their own best position and the knowledge of their Neighborhood's best position.

The standard version of the algorithm can be summarized as follows:

At the beginning, the particles of the swarm have a random repartition in the search space and a random velocity.
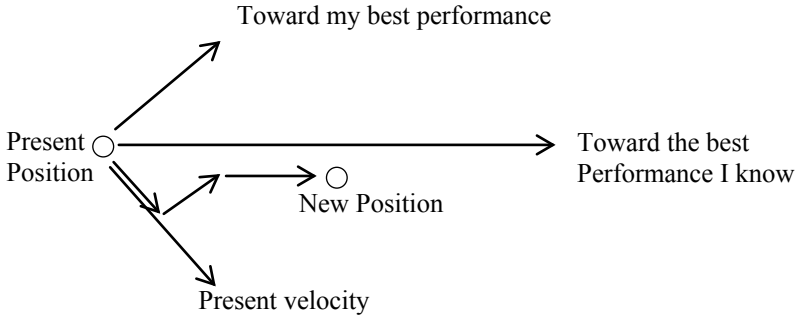
Fig. 2.4 Particle swarm optimization algorithm

Then, at each time step,

- Each particle evaluates the quality of its position and memorizes the best position it has reached at this time and its quality.
- Each particle exchanges information with other particles in its neighborhood in order to know the best performance of each of them.
- At each time step, each particle chooses the best performance it knows and modifies its velocity according to the whole data it has to define its moving as a compromise between three tendencies.
- An adventurous tendency means continuing the journey at the current speed.
- A conservative tendency means going in direction of the best position the particle has currently found.

A panurgian tendency means blindly following the direction toward the optimal point, as pointed by the informants.

2.3.3.2 Particle Swarm Optimization Algorithm

$x_p(t)$: position of the particle p at time t,
$v_p(t)$: velocity of the particle p at time t,
$x_{pi}(t)$: ith component of $x_p(t)$,
$v_{pi}(t)$: ith component of $v_p(t)$,
$x_p^m(t)$: best position the particle p has currently found,
$x_p^M(t)$: best position known by the particle $p \in$ at time t,
$\rho_1(t), \rho_2(t)$ values of random coefficients at time t, $\rho_1, \rho_2 \in [0, 1]$,
$c_1 \leq 1$ constant positive number usually $c_1 \in [0.4, 0.9]$,
$c_{\max} \leq 2$ constant positive number, and
$f(x_p)$ value of the objective function for the particle p at the position x_p.
It comes

$$v_{pi}(t+1) = c_1 v_{pi}(t) + \rho_1 c_{\max}(x_{pi}^M(t) - x_{pi}(t)) + \rho_2 c_{\max}(x_{pi}^m(t) - x_{pi}(t)) \quad (2.16)$$

$$x_{pi}(t+1) = x_{pi}(t) + T(t)v_{pi}(t+1) \tag{2.17}$$

with $T(t) = 1$ or randomly chosen

The stopping test can be a maximum number of iterations or a maximum number of iterations without progress.

2.3.3.3   Bounded Search Space

If due to the algorithm the particle goes out of the authorized search space, we have various possibilities to obtain admissible solution.

$$x_p(t+1) = x_p(t) + \rho T(t)v_p(t+1) \tag{2.18}$$

with $\rho < 1$ till we reach the boundary of the authorized domain

$$x(t+1) = x_p(t) - \rho T(t)v_p(t+1) \tag{2.19}$$

$$\text{if } x_{pi} \in \left[x_{pi\,min}, x_{pi\,max} \right] \tag{2.20}$$

we can choose

$$x_{pi}(t+1) = Min \left\{ Max \left\{ x_{pi}(t) + v_p(t+1); x_{pi\,min} \right\}; x_{pi\,max} \right\} \tag{2.21}$$

2.3.3.4   Selecting the Informants

The number N_p of informants for each particle p is in general limited

$$N_p \in \left[2, \frac{N_p(N_p - 1)}{2} \right] \tag{2.22}$$

by in fact very often $N_p \in [2, 4]$.

The informants' scan be chosen at random and can change if there is no progress when implementing the algorithm.

This algorithm initially defined for continuous problem has been generalized for discrete ones.

2.4   Multi-objective Optimization

2.4.1   Ordered Weighted Averaging (OWA)

After normalization, the various criteria $C_i(x)$ are aggregated in a single one with weight coefficients

$$C_{OWA}(x) = \sum_{i=1}^{n} w_i C_i(x) \tag{2.23}$$

$$\text{with } w_i \in [0, 1], \sum_{i=1}^{n} w_i = 1 \tag{2.24}$$

2.4.2 OWA Using Choquetintegral

It is an OWA-type approach in which the weights w_i are calculated according to the interaction between various criteria.

In order to be self-contained as far as possible, necessary definitions, adapted for multi-criteria decision-making, are given in this section.

Let us consider a finite interval set $N_c = \{1, \ldots, n_c\}$, which can be thought of as an index set of the given criteria [44].

Definition 2.1 A fuzzy measure over $N_c = \{1, \ldots, n_c\}$ is a set function $\mu : P(Nc) \to [0, 1]$, such that

$$\mu(\varphi) = 0, \mu(N_c) = 1 \tag{2.25}$$

$$\mu(A) \leq \mu(B) \text{ whenever } A \subset B \subset N_c \tag{2.26}$$

The meaning attributed to $\mu(A)$ is usually the importance or the power of the coalition A (e.g., for decision-making).

Definition 2.2 Let μ be a fuzzy measure over N_c and $a = (a_1, \ldots, a_{n_c})$ the vector of criteria. The discrete Choquet integral C_μ with respect to μ is defined by

$$C_\mu(a_1, \ldots, a_{n_c}) = \sum_{i=1}^{n_c} (a_i - a_{i-1})\mu(\{i, \ldots, n_c\}) \tag{2.27}$$

with $a_0 = 0$ and $a_1 \leq \cdots \leq a_{n_c}$.

Definition 2.3 Let μ be a fuzzy measure over N_c. The shapely index I_i, for every $i \in N_c$, is defined by

$$I_i = \sum_{k \in N_c - \{i\}} \frac{(n_c - |k| - 1)!|k|!}{n_c!} (\mu(k \cup \{i\}) - \mu(k)) \tag{2.28}$$

where $|k|$ indicates the cardinal of k and $0! = 1$.

Definition 2.4 The average interaction index I_{ij} between two criteria i and j, with respect to a fuzzy measure μ, is defined by

$$I_{ij} = \sum_{k \in N_c - \{i,j\}} \frac{(n_c - |k| - 2)!|k|!}{(n_c - 1)!} \times (\mu(k \cup \{i,j\}) - \mu(k \cup \{i\}) - \mu(k \cup \{j\}) + \mu(k))$$

(2.29)

the interaction index, ranged in $[-1, 1]$, is negative in the case of redundancy, and positive in the case of synergy.

Definition 2.5 The Choquet integral formulation in terms of interaction representation is reduced to an easily interpretable form in the case of (at most) 2-additive measures, which is for any $a = (a_1, \ldots, a_{n_c})$, as follows:

$$C_\mu(a) = \sum_{I_{ij} > 0} (a_i \wedge a_j)|I_{ij}| + \sum_{I_{ij} < 0} (a_i \vee a_j)|I_{ij}| + \sum_{i=1}^{n_c} a_i \left(I_i - \frac{1}{2} \sum_{i \neq j} |I_{ij}| \right)$$ (2.30)

with $\wedge$ and $\vee$ denote min and max, respectively.

2.4.3   Pareto Optimality Approach

Pareto optimality is a measure of efficiency in multi-criteria problems.

In this approach, a non-dominated solution is such that there is no other solution that performs at least as well on every criterion and which is strictly better on at least one of the criteria.

For a Pareto optimal solution, a criterion cannot be improved without damaging at least one of the other criteria.

The set of Pareto optimal solutions corresponds to the Pareto optimal curve also called front of Pareto, Fig. 2.5.

If we have the possibility to determine lower bounds of the various criteria, the Pareto optimality approach can be associated with the OWA approach. After normalization of the criteria, we realize an aggregation of the various criteria with adaptive weights which enables a dynamic search in the direction of the lower bounds point.

For example, if the optimization is realized with a genetic algorithm, Fig. 2.6 represents the evolution of the population.

The OWA approach is the most used method of the multi-objective optimization.

The Choquet integral enables to optimize the choice of the weighted parameters, and the definition of the front of Pareto enables the decider to choose the solution according to his preferences.
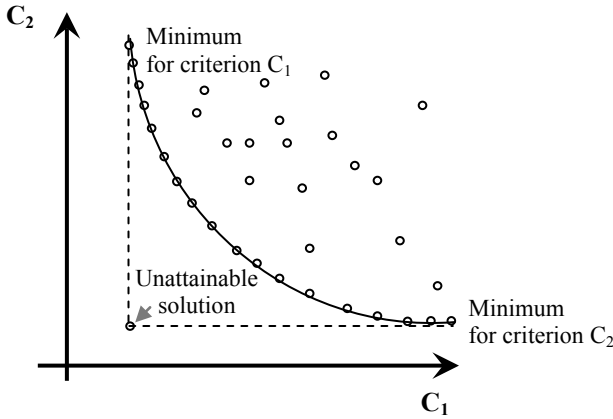
Fig. 2.5 Pareto optimal solution for a problem with two criteria

Fig. 2.6 The evolution of the population
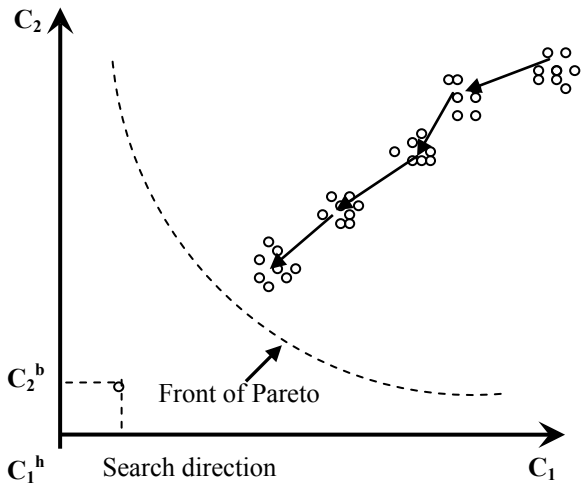
2.5 Conclusion

The various metaheuristics that have been presented here have been implemented in the optimization in manufacturing and control problems but in each case the formulation of the problem has to be adapted to the chosen algorithm.

Very often hybrid approaches are implemented using simultaneously several metaheuristics and usual local search like the hill climbing methods.

References

1. Chipperfield, A.J., Daker, N.V., Whidborne, J.F., Fleming, P.J.: Multi-objective robust control using evolutionary algorithms. In: IEEE International Conference on Industrial Technology, pp. 269–274. Shangai, China (1996)
2. Chipperfield, A.J., Fleming, P.J.: Multi-objective gas turbine engine controller design using genetic algorithms. IEEE Trans. Ind. Electron. **43**(5), 583–587 (1996)
3. Istepanian, R.S., Whidborne, J.F.: Multi-objective design of finite word-length controller structure. In: Proceedings of Congress on Evolutionary Computation, pp. 61–68. Washington DC, USA (1999)
4. Tan, K.C., Lee, T.H., Khor, E.F., Ou, K.: Control system design unification and automation using and incremented multi-objective algorithm. In: Proceedings of 19th IASTED International Conference on Modelling Identification and Control, Innsbruck, Austria (2000)
5. Durate, N., Ruano, A.E., Fonseco, C., Fleming, P.: Accelerating multi-objective control system design using a neuro-genetic approach. In: Congress on Evolutionary Computation, vol. 1, pp. 392–397. IEEE Service Center, USA (2000)
6. Ortmann, M., Weber, W.: Multi-criterion optimization of robot trajectories with evolutionary strategies. In: Proceedings of Genetic and Evolutionary Computation Conference, pp. 310–316. San Francisco California, USA (2001)
7. Stefanoiu, D., Borne, P., Popescu, D., Filip, F.G., El Kamel, A.: Optimization in Engineering Sciences. Wiley, New York (2014)
8. Stefanoiu, D., Borne, P., Popescu, D., Filip, F.G., El Kamel, A., Stefanoiu, D., Borne, P., Popescu, D., Filip, F.G., El Kamel, A.: Metaheuristics – local methods. In: Stefanoiu, D., Borne, P., Popescu, D., Filip, F.G., El Kamel, A. (eds.) Optimization in Engineering Sciences: Approximate and Metaheuristic Methods (2014). https://doi.org/10.1002/9781118648766.ch1
9. Borne, P., Tangour, F.: Metaheuristics for the optimization in planning and scheduling. In: IFAC Proceedings Volumes, vol. 40, Issue 18, pp. 1–7 (2007). ISSN 1474–6670
10. Stefanoiu, D., Borne, P., Popescu, D., Filip, F.G., El Kamel, A., Stefanoiu, D., Borne, P., Popescu, D., Filip, F.G., El Kamel, A.: Metaheuristics – global methods. In: Stefanoiu, D., Borne, P., Popescu, D., Filip, F.G., El Kamel, A. (eds.) Optimization in Engineering Sciences: Approximate and Metaheuristic Methods (2014). https://doi.org/10.1002/9781118648766.ch2
11. Glover, F.: Tabu search, Part I. ORSA J. Comput. **1**, 190–206 (1989)
12. Glover, F.: Tabu search, Part II. ORSA J. Comput. **2**, 4–32 (1990)
13. Oonsivilai, A., Marungsri, B.: Optimal PID tuning for AGC system using adaptive Tabu search. In: Proceedings of the 7th WSEAS International Conference on Power Systems. Beijing, China, September 15–17, 2007 (2007)
14. Ikonomovska, E., Gjorgjevik, D., Loskovska, S.: Using data mining technique for coefficient tuning of an adaptive Tabu search. In: The International Conference on "Computer as a Tool". EUROCON, Warsaw, September 9–1, 2007 (2007)
15. Aytekin, B.: Tabu search algorithm based PID controller tuning for desired system specifications. Int. J. Frankl. Instit. **348**(10), 2012–2795 (2011). December
16. Gharbi, A., Benrejeb, M., Borne, P.: A Taboo search optimization of the control law of nonlinear systems with bounded uncertaities. IJCCC **11**(2), 158–166 (2016)
17. Popescu, D., Gharbi, A., Stefanoiu, D., Borne, P.: Process control design for industrial applications. Minimizing the attractor through Tabu, search, pp. 161–171. Wiley, New York (2017)
18. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)
19. Gonzalez, F., Fernando, L., Aguilar, F., Rita, Q., Gonzalez, G., Alejandro, A.S.: Adaptive simulated annealing for tuning PID controllers. Gildardo AI Commun. **30**(5), 347–362 (2017)
20. Soni, Y.K., Bhatt, R.: Simulated annealing optimized PID controller design using ISE, IAE, IATE and MSE error criteria. Int. J. Adv. Res. Comput. Eng & Technol. (IJARCET) **2**(7) (2013)
21. Lückehe, D., Kramer, O., Weisensee, M.: Simulated annealing with parameter tuning for wind turbine placement optimization. In: CEUR-WS.org, vol. 1458, pp. E16_CRC20 (2015)

22. Alsadiq Y.A.: A comprehensive tuning of distillation column composition controllers using simulated annealing algorithm (SA). In: International Conference on Artificial Intelligence, Energy and Manufacturing Engineering (ICAEME'). Kuala Lumpur (Malaysia), June 9–10, 2014 (2014)
23. Haber, R.E., Haber-Haber, R., del Toro, R.M., José, R.A.: Using simulated annealing for optimal tuning of a PID controller for time-delay systems. an application to a high-performance drilling process. In: International Work-Conference on Artificial Neural Networks, Computational and Ambient Intelligence, pp. 1155–1162. IWANN (2007)
24. Ota, T., Omatu, S.: Tuning of the PID control gains by GA. In: Proceedings 1996, IEEE Conference on Emerging Technologies and Factory Automation. ETFA (1996)
25. Thulasi dharan, S., Kavyarasan, K., Bagyaveereswaran, V.: Tuning of PID controller using optimization techniques for a MIMO process. In: IOP Conference Series: Materials Science and Engineering, p. 263 (2017)
26. Deepa, T., Lakshmi, P.: Comparison of PI controller tuning using GA and PSO for a multi-variable experimental four tank system. Int. J. Eng. Technol. (IJET), **5**(6), 4660–4671. Dec 2013–Jan 2014
27. Moness, M., Moustafa, A.M.: Tuning a digital multivariable controller for a lab-scale helicopter system via simulated annealing and evolutionary algorithms. Trans. Inst. Meas. Control. **37**(10), 1254–1273 (2014)
28. Roeva, O., Slavov, T.: PID controller tuning based on metaheuristic algorithms for bioprocess control. Biotechnol. Biotechnol. Equip. **26**(5), 3267–3277 (2012)
29. Jayachitra, A., Vinodha, R.: Genetic algorithm based PID controller tuning approach for continuous stirred tank reactor. Adva. Artif. Intell. **2014**, 9 (2014)
30. Uren, K.R., VanSchoor, G.: Genetic algorithm based PID tuning for optimal power control of a three-shaft Brayton cycle based power conversion unit. IFAC Proc. Vol. **45**(3), 685–690 (2012)
31. Jaen-Cuellar, A.Y., Rene de J. Romero-Troncoso, R., Morales-Velazquez, L.: PID-controller tuning optimization with genetic algorithms in servo systems. Int. J. Adv. Robot. Syst. **10**, 324–2013 (2013)
32. Holland, J.H.: Genetic algorithms and the optimal allocations of trials. SIAM J. Comput. **2**, 88–105 (1973)
33. Hanifah, R.A., Toha, S.F., Ahmad, S.: PID-ant colony optimization (ACO) control for electric power assist steering system for electric vehicle. In: IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA). (2013)
34. Nagaraj, B., Murugananth, N.: A comparative study of PID controller tuning using GA, EP, PSO and ACO. International Conference on Communication Control and Computing Technologies (2010)
35. Chiham, I., Liouane, N., Borne, P.: Tuning PID controller using multiobjective ant colony optimization. Hindawi Publ. Corp. Appl. Comput. Intell. Soft Comput. **2012**, Article11 (2012)
36. Kaliannan, J., Baskaran, A., Dey, N., Ashour, A.S.: Ant colony optimization algorithm based PID controller for LFC of single area power system with non-linearity and boiler dynamics. World J. Model. Simul. **12**(1), 3–14 (2016)
37. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: IEEE International Conference on Neural Networks, pp. 1942–1948. Piscataway, NJ, Proc. (1995)
38. Kennedy, J., Eberhart, R., Shi, Y.: Swarm Intelligence. Morgan Kaufmann Academic Press (2001)
39. Eswaran, T., SureshKumar, V.: Particle swarm optimization (PSO)-based tuning technique for PI controller for management of a distributed static synchronous compensator (DSTATCOM) for improved dynamic response and power quality. Journal of Applied Research and Technology **15**(2), 173–189 (2017)
40. Iwan Solihin, M., Fook Tack, L., Leap Kean, M.: Tuning of PID controller using particle swarm optimization (PSO). In: Proceeding of the International Conference on Advanced Science, Engineering and Information, Technology. Malaysia, Jan 14–15, 2011 (2011)
41. Li, X., Yu, F., Wang, Y.: PSO algorithm based online self-tuning of pid controller. In: International Conference on Computational Intelligence and Security (CIS 2007), Dec 15–19, 2007 (2007)

42. Freire, H.F., de Moura Oliveira, P.B., Solteiro Pires, E.J., Bessa, M.: Many-objective PSO PID controller tuning. In: Proceedings of the 11th Portuguese Conference on Automatic Control, pp 183–192. CONTROLO'2014 (2014)
43. Vincent, A.K., Nersisson, R.: Particle swarm optimization based PID controller tuning for level control of two tank system. In: 14th ICSET-2017 IOP Conference Series: Materials Science and Engineering, p. 263 (2017)
44. Choquet, G.: Theory of capacities. Annales de l'Institut Fourier **5**, 131–295 (1953)

Chapter 3
Optimal Controller Parameter Tuning from Multi/Many-objective Optimization Algorithms

O. Tolga Altinoz

Abstract Controller performance is evaluated with the properties of steady-state and transient response of the system at the time domain. The compensators and conventional controllers like PID are designed, so that the desired performance is reached only by adjusting the controller parameters; this adjustment mechanism is called tuning. Even many approaches are proposed for tuning; still, it remains one of the problems of control theory due to the imperfect modeling, disturbance, and problem complexity. At this stage, optimization algorithms have helped the researchers to find these parameters via designing objective function concerning the time response characteristics of the system. As the expectations (for example, steady-state error, overshoot, rise time, settling time) related to the system performance are increased in number, the number of objectives is also increased. As a result, multi-objective optimization algorithms have applied to solve these problems. In this chapter, a set of benchmark tuning problems is defined as the test—benchmark—problems. Then, by using the multi- and many-objective optimization algorithms, the performance of the controlled system with respect to the Pareto approximate set is compared with each other. The multi/many-objective optimization algorithms evaluated in this chapter are as follows: Multi-objective Evolutionary Algorithm based on Decomposition (MOEAD), Non-dominated Sorting Genetic Algorithm II (NSGA-II), Multi-objective Particle Swarm Optimization (MOPSO), Strength Pareto Evolutionary Algorithm 2 (SPEA2), Approximation-guided Evolutionary Algorithm II (AGE-II), and Reference Vector Guided Evolutionary Algorithm (RVEA).

3.1 Introduction

The controller algorithms are defined to get the desired response from simple or complex systems: from a basic spring system to more complex space shuttles. The controller algorithms are designed for the given design specifics and output

O. T. Altinoz (✉)
Department of Electrical and Electronics Engineering, Ankara University, Ankara, Turkey
e-mail: taltinoz@ankara.edu.tr

properties. Among many control algorithms—or simply controllers—the most fundamental controllers are called PID controllers. The Proportional–Integral–Derivative (PID) controllers are one of the most basic and old algorithms, but still, it is efficiently applied to many real-life industrial problems. The performance of the overall system with a PID controller is greatly depended on the accuracy of the adjusted parameters of the PID controller.

The PID controller contains three terms that should be adjusted concerning the desired performance. These parameters are named as proportional, integral, and derivative terms. These three terms have a direct effect on the performance of the system, where these three terms may affect each other concerning the given system. However, in general, the steady-state error can be reduced or even eliminated with an integral that makes this term as an important control parameter. The derivative term may change the transient response of the system, but it may cause instability. PI gives relatively lower overshoot with a small settling/rise time when compared to an ideal integral compensator. To compare the performances of PI and PID controller, Jagatheesan and Anand [1] show that for only I-ideal integral compensator–controller, the overshoot becomes relatively larger than PI-ideal proportional and integral controller, which is also supported by Laghardi et al. [2]. They present a toolset that shows PID and PI controllers which give almost the same overshoot performances, and also the integral term occurs as the most important term to almost cancel the steady-state error. For a desired time or frequency response, the accurate values for controller parameters have to be determined. Therefore, the method or algorithm which is designed to find accurate parameters for PID (PI, PD, or PID) controllers is called tuning. The well-known tuning algorithms are named as Ziegler–Nichols [3], Cohen-Coon [4], and Lambda [5]. These methodologies are based on a set of rules that should be followed to find the acceptable—not optimal—parameters for the predefined limited number of controller structures like feedback and feedforward controller. These methodologies poorly choose the parameters for stability and frequently present oscillation at the output of the overall system. The parameter selection rule sets are generally depended on reducing the error between set point and the output, as fast as possible. However, the intelligent methodologies can be evaluated as other properties of the overall system like overshoot and rise time. Therefore, among many tuning algorithms, intelligent methods such as optimization algorithms improve both the steady-state error and transient response properties.

Initially, the intelligent methods are applied to the system as control signal generators. Instead of PI or PID controllers, a pre-prepared intelligent structure (generally a learning-based architecture) is applied instead of a controller. As an intelligence methodology, fuzzy logic, neural network, and a joint method named neuro-fuzzy have applied to PI controller framework of the study of Ram and Jha [6]. However, the reported performance is almost the same as each other where the rise time and settling time reduce. Similarly, another neuro-fuzzy system named Adaptive Neuro-Fuzzy Inference System (ANFIS) is discussed by Khuntia ve Panda [7]. The results support the intelligent controller performance against the conventional compensators. Meanwhile, as the impact of the computational (especially the evolutionary algorithms) optimization algorithms is increased, they have applied to find the controller

parameters as an optimal toolset. The selection of many different objective functions makes the tuning algorithm more flexible. Also, the optimal parameter selection does not only improve the performance but also it increases the robustness of the system. Singh et al. [8] showed that the performance of the overall system depends on the appropriate selection of a parameter, and it is shown that an optimal parameter selection for PI controller has increased the performance using the Genetic Algorithm (GA). Nanda et al. [9] discussed the performance of pure-I, PI, PID, and ID controllers which are tuned using intelligent methods on the real-life problem, where it is observed that the derivative term increases the noise of the overall system. Similarly, PID controller is tuned using Particle Swarm Optimization (PSO) algorithm [10, 11], GA [12], and Differential Evolution (DE) [13], at the study of Dangor et al. in [14]. Similarly, nature-inspired optimization algorithm Gravitational Search Algorithm (GSA) [15] is applied for the tuning problem. The results are concluded that almost the same performance is obtained from all of these algorithms. Like single-objective optimization algorithms, multi-objective optimization algorithms are also applied to find the proper selection of controller parameters. Tseng et al. [16] and [17] have proposed a study that the sufficient performance of the multi-objective PID control design is obtained under the plant uncertainties and external disturbance. In addition to these disturbances, the parametric uncertainties are also considered.

Similarly, Tang et al. [18] are applied Multi-objective Genetic Algorithm (MOGA) for fuzzy PID controller. As a different perspective, frequency-domain optimal tuning PID and multi-objective tuning PID controllers are discussed by Liu and Daley [19] for the industrial problems. In [20] and [21], a multi-objective optimization algorithm (NSGA-II) is applied without using scalarization functions for the PID tuning algorithm, and relatively better results are obtained compared to the previous similar studies.

In this chapter, like the previous studies, multi-objective optimization algorithms are applied to PID and lead compensator to get a proper controller parameter set. However, in addition to this study, many-objective optimization algorithms are also applied to the problem. For this purpose, three-benchmark system is defined as test problems. These systems are selected from the real-life problems. They are the ball and beam system, heat exchanger, and distillation column.

Moreover, four different controller structures are applied to these problems. Also, six different optimization algorithms are implemented in four different problem cases. These algorithms are Multi-objective Evolutionary Algorithm based on Decomposition (MOEAD), Non-dominated Sorting Genetic Algorithm II (NSGA-II), Multi-objective Particle Swarm Optimization (MOPSO), Strength Pareto Evolutionary Algorithm 2 (SPEA2), Approximation-guided Evolutionary Algorithm II (AGE-II), and Reference Vector Guided Evolutionary Algorithm (RVEA). The performance of these algorithms is compared with each other concerning the controller parameters.

This chapter is organized as follows: Section 3.2 gives explanations of the real-life benchmark problems for this study. In Section 3.3, optimization algorithms are explained briefly. In Section 3.4, the implementation and its steps are explained. Finally, the last section concludes the chapter.

3.2 Benchmark Problems

In this section, three tuning problems are selected as test problems of the current research. These three systems have different properties with a different set of controller approach. Ball and beam system is one of the fundamental problems in control engineering. The system is defined by a set of the Ordinary Differential Equations (ODE). The PID or lead compensator can be easily applied to get the desired solution. The second problem is a common industrial system called a heat exchanger. Heat exchanger is also one of the fundamental devices used in industry to cooldown or heat the gas or liquid inside a tank. The level of gas or liquid in the tank changes with the input and output. The change of the temperature is modeled as a disturbance. An additional controller—feedforward controller—is added to the system to handle this disturbance. The last example is from a chemical process called distillation column. In that system, two different liquids or gases are mixed, and the concentration at each tank is observed to get the desired mixture. Therefore, the system has two outputs with two inputs, which is a multi-variable system. Two feedback controllers for each input and a single gain are defined for the system. In this section, these three test systems are explained and defined.

3.2.1 Ball and Beam System

Ball and beam system—BBS—(Figure 3.1) is formed as a balancing problem by adjusting the angle of the beam. The ball is free to move on the beam, and this problem aims to hold the position of the free moving ball to the desired location on the beam. This location is generally the midpoint of the beam. Since the movement of the ball is only among the beam and it is not possible to change the direction of the ball to other dimensions, this problem can be considered as a single Degree of Freedom (DOF) problem. The equilibrium motion of the ball is expressed as follows:

$$mr\dot{\alpha}^2 = mgsin\alpha + \left(\frac{J}{R^2} + m\right)\ddot{r}, \tag{3.1}$$

Fig. 3.1 Ball and Beam system
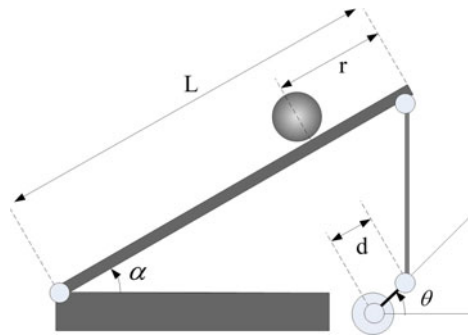
where m is the mass of the ball, R is the radius of the ball, d is the level arm offset due to the mechanical and sensor connection, g is the gravitational constant, L is the length of the beam, J is the moment of inertia of the ball, r is the ball's position, θ is gear angle, and finally $\alpha = \frac{d}{L}\theta$ is the beam angle.

In this system, the controller produces the action signal to the actuator which is connected to the beam. The actuator changes the beam angle. In this way, the free rolling ball changes its position. The sensor on the beam gives the almost exact position of the ball. In this chapter, the system parameters are selected as $m = 10$ g $R = 1.5$ cm, $d = 3$ cm, $g = -9.8$ m/s^2, $L = 1$ m, and $J = 10^{-5}$ g/m^2.

3.2.2 Stirring Tank with Heat Exchanger

Stirring tank with heat exchanger—or simply Heat Exchanger—is a system composed of one tank with physically connected two inputs and one output. It is desired to maintain the temperature inside the tank is fixed. A top inlet transfers gas or liquid—inflow (as shown in Fig. 3.2)—to the tank with a various temperature which depends on the environmental temperature or the temperature of the source—variation in the inflow temperature. This changeable temperature at the input is modeled as the disturbance model. The aim is to hold the temperature inside the tank to a certain degree by the aid of incoming stream or may be a liquid that wraps the tank to set the desired temperature. In general usage of a heat exchanger system, the aim is to increase/decrease the incoming temperature of the liquid with the aid of adjustable density of hot steam. The density of the hot steam is adjusted using a valve. Figure 3.2 gives the graphical demonstration of the heat exchange system.

The inflow liquid changes the temperature inside the tank. Since the temperature at the inflow liquid is not constant, the change at the temperature inside the tank is variable. Then, this change due to the variable temperature of the inflow liquid is modeled as a disturbance. Therefore, the valve must be controlled so that the temperature inside the tank must remain at a constant value. In reality, the stirring tank is generally huge, and many temperature sensors are needed to get an accurate temperature—with sensor fusion. However, for simplicity, it is assumed that the

Fig. 3.2 Stirring tank with heat exchanger
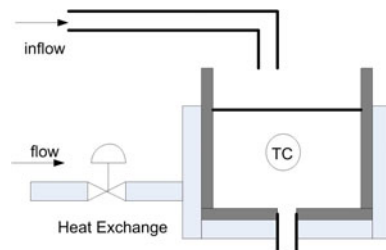
temperature inside the tank is homogeneous and only a single sensor is used in this system.

The temperature inside the tank is a summation of two models: the disturbance model and heat exchanger model. These two models have the same characteristics related to the change of the temperature. Initially, the change at the temperature is almost zero, and after some time, the temperature begins to increase and increase exponentially. The change of the temperature can be defined as a delay time and a simple ODE. Therefore, both models are defined as the first-order system with a relatively large delay. Hence, the mathematical relation pattern is defined as

$$e^{-t_1 s} \frac{1}{Ts + 1}, \tag{3.2}$$

where T is the time constant and t_1 is the delay time. For the implementations of this chapter, $T = 35, t_1 = 25$ are selected for disturbance model, and $T = 21, t_1 = 15$ are selected for heat exchanger model. As a result, the sum of two controller actions is needed to get a desired voltage level of the valve.

3.2.3 Distillation Column

Originally, distillation is a process to divide a joint component into two—or more—original components by heating them to a certain degree, a separation process. The distillation column is a system that made of a couple of plates from under the tank to the top of it. The temperature decreases at the top of the column. Heaviest components remain at the bottom and lightest at the top. The joint liquid enters from one point of the column, and it is vaporized. The bottom of the column contains the liquid with a higher boiling temperature that is one of the original liquids. Figure 3.3 gives the graphical demonstration of the distillation column.

The basic principle of the distillation column (Figure 3.3) is the resupply mechanism. The liquid at the base (about B) is heated and repumped to the column, and these are boiled up; some of them remains as one of the outputs. The liquid vaporizes as moving to the top of the column. At the top (about D), some of the vaporized liquid is returned and resupplied to the column as reflux; the remaining liquid is the—distillate—product.

The system is given in Figure 3.3 as an example of Multi-Input and Multi-Output system (MIMO), where f_L—reflux flow—and f_V—steam flow—are the inputs, and f_D and f_B are the outputs from two heat exchanger systems B and D, respectively. Hence in total, there are four relations between each input and output. Since the relation at each exchanger is defined as temperature relation, and they are heat exchangers, these systems are modeled as the first-order transfer function for all four relations (Note that these four relations are between inputs and outputs. For each input, there are two transfer functions for outputs).

Fig. 3.3 Distillation column
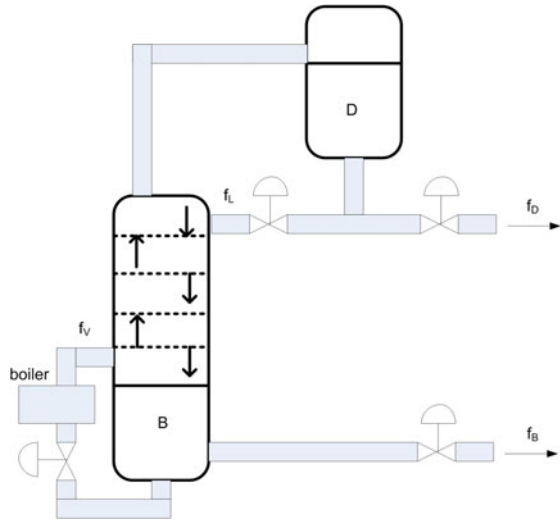
$$\frac{T_1}{T_2 s + 1},\tag{3.3}$$

where for this chapter for all relations $T_2 = 75$, and for f_L to f_D $T_1 = 88$, for f_L to f_B $T_1 = 108$, for f_V to f_D $T_1 = -86$, and finally for f_V to f_B $T_1 = -106$. Therefore, the transfer function matrix is defined as given below:

$$\begin{bmatrix} \frac{88}{75s+1} & \frac{108}{75s+1} \\[2mm] \frac{-86}{75s+1} & \frac{-106}{75s+1} \end{bmatrix}.\tag{3.4}$$

3.3 Optimization Algorithms

In this research, six multi/many-objective optimization algorithms are evaluated to optimize the controller parameters. In this section, these algorithms are explained in brief.

3.3.1 Strength Pareto Evolutionary Algorithm II

Strength pareto evolutionary algorithm II—SPEAII—is the improved version of the SPEA that has proposed by Zitzler and Laumans in 2001 [22]. The algorithm begins with the initialization of the population randomly on the decision space,

and it evaluates the solution candidates with rank-based dominance principle. This algorithm uses an elitist strategy so that an external archive is formed with best members and new solutions are produced from this archive set. Also with a density estimator inside the algorithm, the solution inside the archive set maintains a spread density. This empty archive set is defined for the usage at each iteration. The size of the archive matrix must be same as the number of population. Therefore, the members of the archive will be less than or equal to the size of the population. At each generation—iteration—the objective function values are calculated for the current population and the archive members, if any. The domination principle is applied to the solution candidates, and members are sorted concerning the dominance principle. These solution candidates are joined to a single set, and they are sorted. Next, dominated solutions are deleted from this set. This operation is called truncation operator. If the termination conditions are not met, then the obtained population is applied to tournament selection operator. As the last steps, crossover and mutation operators are applied, and the survived members continue on the next generation.

3.3.2 Non-dominated Sorting Genetic Algorithm II

The Non-dominated Sorting Genetic Algorithm II (NSGA-II) is proposed by Deb et al. in 2002 [23]. NSGA-II also begins with the initialization of the randomly selected chromosomes—members. At the beginning of each generation—iteration— the overall population is changed by mutation and combination operators, and they are stored as a new vector. Then, these two vectors are joined with each other, and they are sorted concerning the objective values. In the sorting phase, each of the members is assigned to rank concerning the Euclidean distance between each other, with an operator called the crowding distance. As the final step, the best chromosomes are selected concerning their rank beginning from the first front and crowding distance values. Lower ranked chromosomes survive to the next generation based on their crowding distance. This process is repeated until the termination conditions are met.

3.3.3 Multi-objective Particle Swarm Optimization

Multi-objective Particle Swarm Optimization (MOPSO) was proposed by Coello et al. in 2004 [24, 25]. In MOPSO, as the multi-objective optimization, it is expected to get a set of solutions at a single iteration. Therefore, the algorithm begins with a population. In the single-objective PSO, the swarm follows the leader who guides the swarm on decision space. At the multi-objective case, more than one leader is needed that followed by different members. Therefore, like SPEA2, an external archive is defined to store the leaders. This set contains a non-dominated solution among all of the iterations. The leaders are selected from this set. The idea

is to lead the solutions to the Pareto front and maximize the distribution of the solutions on the front [26]. The MOPSO algorithm begins with the random initialization of the position and velocity of the members. Also, temporary data which holds the leader set and non-dominated members is initialized as an empty matrix. Then, the population is sorted based on the dominance idea. The best members are selected as the leaders of the population, and they are stored in a matrix. As the update phase, positions and velocities are altered by using the same formulation of single- objective PSO, given as follows:

$$v_i^{(k+1)} = wv_i^{(k)} + c_1 rand \left(p_{Best,i}^{(k)} - x_i^{(k)} \right) + c_2 rand \left(g_{Best}^{(k)} - x_i^{(k)} \right) \qquad (3.5)$$

$$x_i^{(k+1)} = x_i^k + v_i^{(k+1)}, \qquad (3.6)$$

where x is the position, and v is the velocity of each member i. p_{Best} gives the position of the personal best, and position of the best particle member is given as g_{Best}. The inertia weight w is defined to control the impact of the previous solution, and the learning factors c_1 and c_2 are the algorithmic constants. The last step of the algorithm is the mutation operator. For the next iteration, only the best members are survived, and these processes are repeated until the algorithm is terminated.

3.3.4 Multi-objective Evolutionary Algorithm Based on Decomposition

Multi-objective Evolutionary Algorithm based on Decomposition—MOEAD—is proposed by Zhang et al. in 2007 [27]. The decomposition idea is applied to convert the many-objective problem into a couple of single-objective subproblems. The idea is to solve these single-objective subproblems using the objective values of the neighborhood solutions. The decomposition is a process similar to scalarization. Hence, both weighted sum and Tchebycheff approaches are integrated into the algorithm as the so-called decomposition process. The algorithm begins with the initialization of reference points, related decomposition parameters, and randomly distributed initial positions. Next, Euclidean distances between any of the two solutions are calculated. Then, by using the evolutionary operators, new solutions are generated. In the next step, among all of the neighbor solutions of the current member, if a better solution is obtained from the neighborhood subproblem, this solution is replaced with the current member's solution. Then the next generation begins. This process is repeated until termination conditions are met.

3.3.5 *Approximation-guided Evolutionary Algorithm II*

Approximation-guided Evolutionary Algorithm II—AGE2—is introduced by Bringmann et al. in 2013 [28]. AGE is another Evolutionary Algorithm (EA) such that it evaluates the genetic operators to alter the solutions or search the solution space. This algorithm also begins with the randomly initialized populations, and a new set is defined as an empty matrix. Then, two random members are selected from the population and applied to crossover and mutation. Finally, a new population of offspring is obtained. From each of these offsprings, the domination principle is applied, and only non-dominated members remain. Next, the current population and offspring are joined with each other. Finally, as the selection operator, an introduced method called "approximation quality" of this joint population is measured and sorted. Then, the best members are survived to the next generator.

3.3.6 *Reference Vector Guided Evolutionary Algorithm*

Reference Vector Guided Evolutionary Algorithm—RVEA—is proposed by Cheng et al. in 2016 [29]. Like MOEAD problem, RVEA is another decomposition-based many-objective optimization algorithm. It evaluates reference vectors to decompose the original problem into many single-objective subproblems. A new scalarization approach, named the angle-penalized algorithm, is proposed to decompose the problem. This optimization algorithm is EA-based algorithm such that offspring is calculated from EA operators. These offspring members are joint with the current population. Next, the best members are selected from the operator called reference vector guided selected. As the last step, these selected members are adapted to the search environment. These steps are repeated until the termination conditions are met.

3.4 Implementation

Three real-life problems are selected as test problems to find the proper controller parameters. For this purpose, six different multi- and many-objective optimization algorithms are evaluated. The details of the problems related to the control algorithms are presented as subsections. About the optimization algorithms, to make an efficient and fair comparison, the number of populations for each of the optimization algorithms is selected as 10^2, and a maximum number of function evaluations for each of the optimization algorithms is selected as 5×10^3. Each of these algorithms is executed as 30 free runs. The search space is inside from [0, 1] for all of the problems. The mean and standard deviation of the selected metrics are numerically given

Fig. 3.4 Feedback control system for ball and beam system
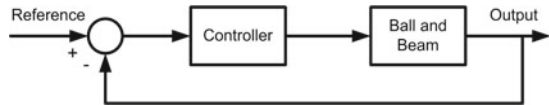
for comparison. Also, the mean distribution of the solutions and both objective space and decision space is graphically presented.

3.4.1 Control Algorithms

In this chapter, three benchmark models are proposed as test problems. These models (or systems) have different properties. Therefore, different control algorithms are needed to get the desired performance. Figure 3.4 gives the overall control model for BBS. Two different controllers are selected for this model. Both PID and lead compensator are applied to the problem. The mathematical expressions for PID controller are presented as

$$G_c(s) = K_P + \frac{K_I}{s} + K_D s, \tag{3.7}$$

where K_P, K_I, and K_D are three parameters of the PID controllers which are the proportional term, Integral term, and derivative term, respectively. The BBS with PID controller becomes the first problem of this research, named as Case-1—$C1$—.

In addition to the PID controller, as a part of this research, the lead compensator is also applied to the BBS, named as Case-2—$C2$—. The mathematical expression of this controller is as follows:

$$G_c(s) = K_1 \frac{s + K_2}{s + K_3}. \tag{3.8}$$

The second system is called the heat exchanger system. Heat exchanger system is a conventional industrial device that used for holding the temperature inside a tank by adjusting the incoming heat. The block diagram of the heat exchanger is given in Fig. 3.5.

Heat exchanger contains two different model blocks which are disturbance model and system model. The disturbance model is defined for the incoming liquid that changes the temperature inside the tank. The temperature in the tank varies due to the imperfect flow of the liquid with a different temperature. The disturbance has a direct effect on the output of the system. For the heat exchanger system, two PID controllers are defined, and their six different parameters are optimized with the given algorithms. The feedforward controller structure is named as Case-3 $C3$.

The last problem is called distillation column. This problem is a standard example of the Multi-Input-Multi-Output (MIMO) system in the industry especially chemical
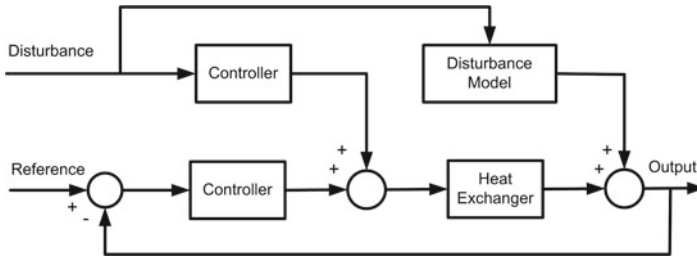
Fig. 3.5 Feedforward—feedback control system for heat exchanger system

Fig. 3.6 Multi-variable control system for distillation column system
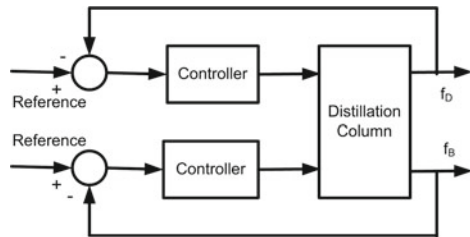
reactions. Figure 3.6 shows the graphical representation of the system. This system has two inputs and two outputs. In total four different relations exist in this system. As a result, two controllers are defined for the problem. As a difference instead of PID controllers, two PI controllers and a single common proportional term are selected as a controller. Therefore, in total five unknowns are optimized to get the best performance. This problem is named as Case-4 $C4$.

3.4.2 Objective Functions

For all computational optimization algorithms, the aim is to reach the desired problem at a given level. In other words, to get the minimum or maximum of the defined objective function (sometimes called cost function). Therefore, the objective function has a direct influence on the performance of the problem. In this study, two objectives are defined as the aims of the optimization algorithms. These functions are defined as follows:

$$f_1 = \int_{t_0}^{t_f} |e(t)| \, dt \qquad (3.9)$$

$$f_2 = t_r + t_s + t_p, \qquad (3.10)$$

where t_0 and t_f are initial time and final time of the problem, respectively. The error e between the desired reference input and the output of the system. the transient response properties t_r, t_s, and t_p correspond to the rise, setting, and peak time of the system, respectively. Since the multi-variable system (distillation column) contains two outputs, these two objective functions are modified as the sum of each output. Hence, modified objective functions are given as follows, where the superscript corresponds to the index of the input.

$$f_1 = \int_{t_0}^{t_f} |e_1(t) + e_2(t)| \, dt \tag{3.11}$$

$$f_2 = t_r^1 + t_s^1 + t_p^1 + t_r^2 + t_s^2 + t_p^2. \tag{3.12}$$

3.4.3 Results

The three systems with four different controller actions are optimized with six optimization algorithms 30 independent times. Initially, to fairly compare the performance of these algorithms, the functions that convert the performance into numerical values are defined. They are called metrics. Since the results are defined in the objective space, it is not possible to mention about only one solution. Instead, two solutions form a shape on the objective space. Therefore two properties of these shapes are evaluated with two different metrics. The first one is for the distribution of the results on the objective space. Therefore, spacing metric [30] is selected as one of the metrics that measure the distribution of the solutions on the objective space. The formulation of this metric is given as follows:

$$M_1 = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (d_i - d_m)^2}, \tag{3.13}$$

where d is the minimum of the sum of the objective values as distance measurement, and d_m is the mean of the distance measure. This value indicates the distribution of the approximate Pareto solution on the objective space such that for a smaller value corresponds to better distribution.

Second, as a part of this research, to evaluate the accuracy—or closeness—of the solutions concerning the objective values, a basic method is selected as metric since the true Pareto front is not known. This metric is the average of the distance to the origin at the objective space. Therefore, this metric is formulated as

$$M_2 = \frac{1}{n} \sum_{i=1}^{n} (E_i), \tag{3.14}$$

Table 3.1 Distribution -spacing metric M_1- of the result obtained from optimization algorithms

Problem	NSGAII	MOPSO	SPEA2	AGEII	MOEAD	RVEA
C1	5.6291e-2 (7.14e-2)	7.1939e-2 (9.36e-2)	6.0348e-2 (5.96e-2)	6.8882e-4 (8.18e-6)	1.0208e-4 (1.70e-5)	2.4931e-2 (1.06e-2)
C2	6.8736e-3 (2.24e-4)	8.1293e-3 (1.81e-3)	4.6489e-3 (2.79e-3)	4.5584e-2 (3.14e-2)	4.1463e-3 (1.05e-2)	3.3767e-2 (1.71e-2)
C3	2.0573e-1 (2.33e-1)	3.3908e-1 (4.45e-1)	3.4867e-1 (3.62e-1)	4.0019e-2 (5.17e-2)	1.2535e-2 (2.49e-3)	2.7170e-1 (3.06e-1)
C4	1.2023e+4 (1.70e+4)	8.3789e-2 (1.18e-1)	9.2825e-2 (3.05e-2)	1.9311e-1 (1.01e-1)	8.3840e-2 (5.70e-2)	1.0570e-1 (2.78e-2)

**Table 3.2** Closeness -M_2- of the result obtained from optimization algorithms

Problem	NSGAII	MOPSO	SPEA2	AGEII	MOEAD	RVEA
C1	10.7254 (1.12e-1)	10.8968 (9.8e-2)	11.3532 (8.5e-2)	10.6400 (2.56e-1)	10.6244 (5.90e-2)	10.8018 (5.6e-1)
C2	9.1722 (1.01)	9.2390 (0.98)	9.3862 (1.9)	10.0612 (1.5)	8.7949 (1.02)	8.9892 (2.39)
C3	21.5386 (2.45)	22.4757 (3.06)	21.7749 (1.98)	22.0225 (2.03)	21.6681 (1.67)	21.8433 (1.53)
C4	112.8853 (10.15)	146.2276 (5.46)	157.1626 (12.02)	120.8913 (10.10)	120.7414 (5.00)	112.2809 (13.07)

where E is the Euclidean distance of each point at the objective space to the origin and n is the total number of points at the objective space. Hence, the smallest metric value corresponds to a better performance.

Table 3.1 performs the optimization algorithm concerning the first metric related to the distribution. The mean values are given inside all of the entry, and in parenthesis, the standard deviation of these results are reported. From this table, it is clear that the well-distributed solutions are obtained from the MOEAD algorithm. However, for the problem C4, MOPSO and MOEAD algorithms present almost the same metric values. However, MOPSO gives the worst performance for C1 and C2. Also, NSGA2 gives the worst performance for the C4 problem, where there is a huge gap between the other results.

Table 3.2 gives the statistical results for the second metric. From this table, the performance of MOEAD can be observed for problems C1 and C2, which are the best. Even, NSGA-II gives the best performance for the C3 problem. Still, MOEAD presents acceptable results. However, RVEA gives the best result for C4 with NSGA-II algorithm. The corresponding controller parameters are graphically demonstrated at the decision space in Figures 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13 and 3.14.

Figures 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13, and 3.14 give both objective space and decision space distributions of the solution for all four cases. When these figures are investigated, it can be observed that it is possible to get a proper controller parameter set for C1, C2, and C3. However, for C4, the algorithms produced more spread solutions on a relatively larger area with relatively bigger metric values.
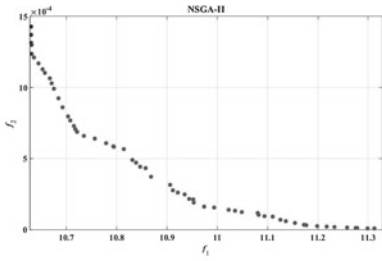
(a) NSGA2 Objective space two objective PIDs for BBS
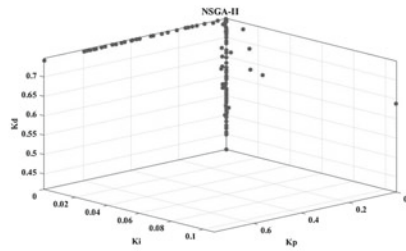
(b) NSGA2 Decision space
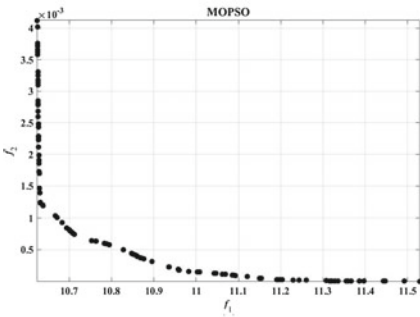
(c) MOPSO Objective space two objective PIDs for BBS

(d) MOPSO Decision space

(e) SPEA2 Objective space two objective PIDs for BBS

(f) SPEA2 Decision space

Fig. 3.7 Pareto approximates front and decision space (controller parameters) obtained from NSGA2, MOPSO, and SPEA2 algorithms for BBS problem
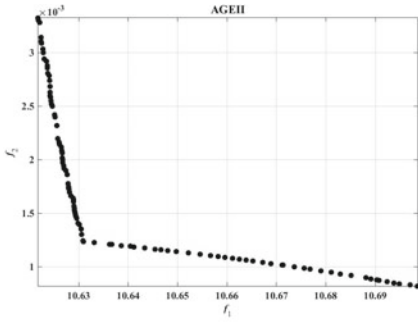
(a) AGE2 Objective space two objective PIDs for BBS

(b) AGE2 Decision space

(c) MOEAD Objective space two objective PIDs for BBS

(d) MOEAD Decision space

(e) RVEA Objective space two objective PIDs for BBS
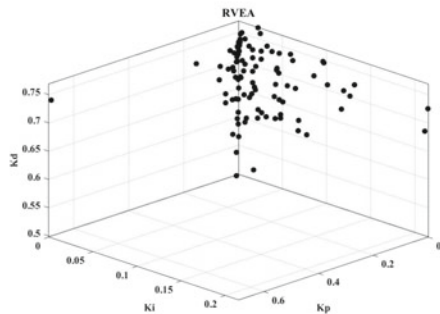
(f) RVEA Decision space

Fig. 3.8 Pareto approximates front and decision space (controller parameters) obtained from AGE2, MOEAD, and RVEA algorithms for BBS problem
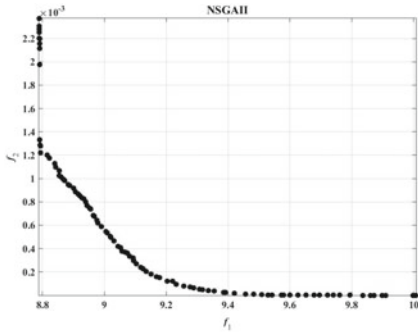
(a) NSGA2 Objective space two objective Compensator for BBS
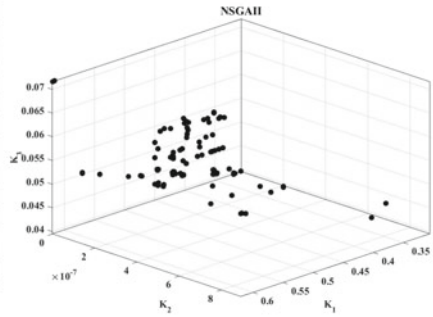
(b) NSGA2 Decision space
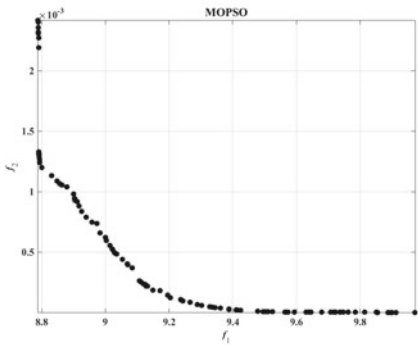
(c) MOPSO Objective space two objective Compensator for BB
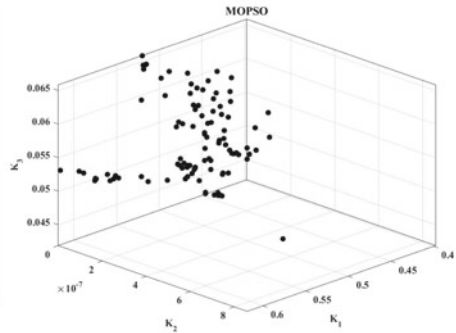
(d) MOPSO Decision space
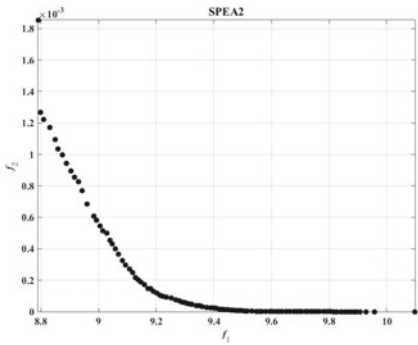
(e) SPEA2 Objective space two objective Compensator for BBS
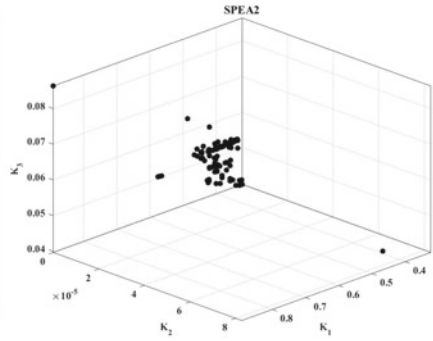
(f) SPEA2 Decision space

Fig. 3.9 Pareto approximates front and decision space (controller parameters) obtained from NSGA2, MOPSO, and SPEA2 algorithms for BBS problem
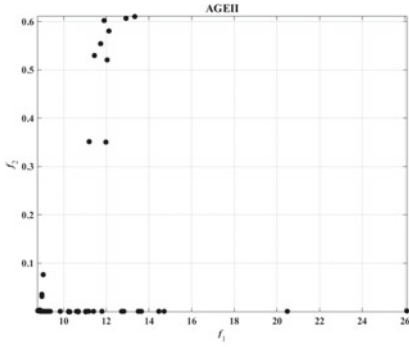
(a) AGE2 Objective space two objective Compensator for BBS
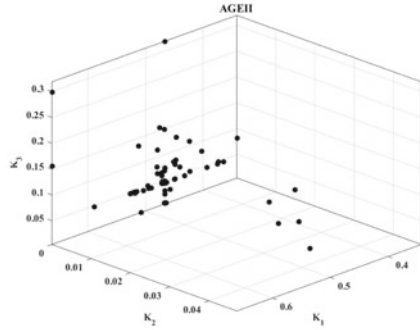
(b) AGE2 Decision space
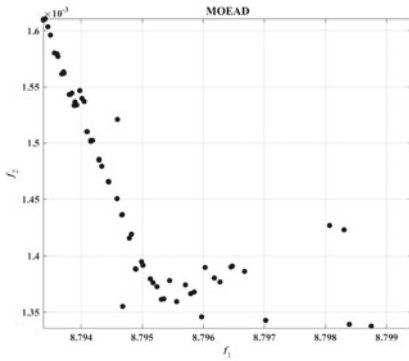
(c) MOEAD Objective space two objective Compensator for BBS

(d) MOEAD Decision space

(e) RVEA Objective space two objective Compensator for BBS
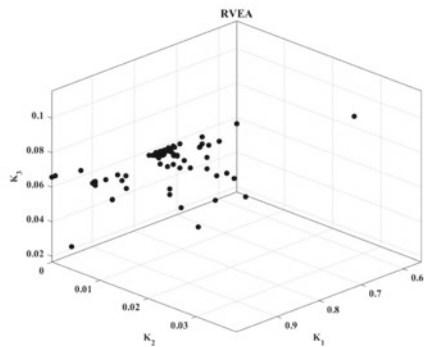
(f) RVEA Decision space

Fig. 3.10 Pareto approximates front and decision space (controller parameters) obtained from AGE2, MOEAD, and RVEA algorithms for BBS problem
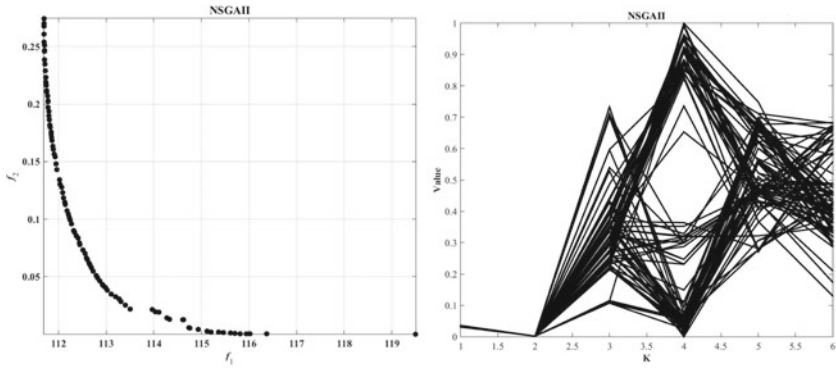
(a) NSGA2 Objective space two objective PIDs for Heat Exchanger

(b) NSGA2 Decision space
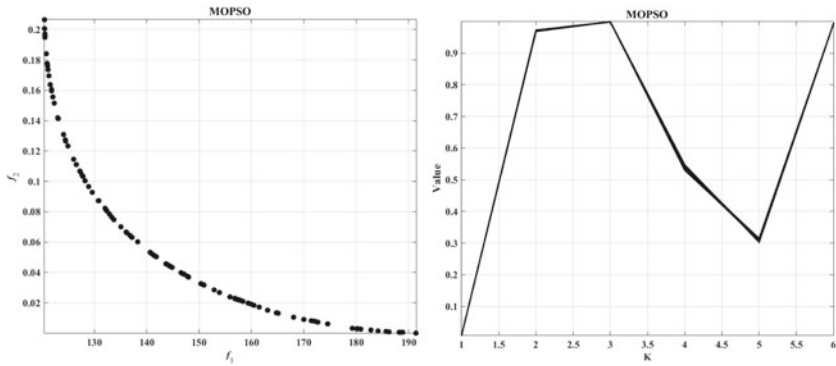
(c) MOPSO Objective space two objective PIDs for Heat Exchanger

(d) MOPSO Decision space
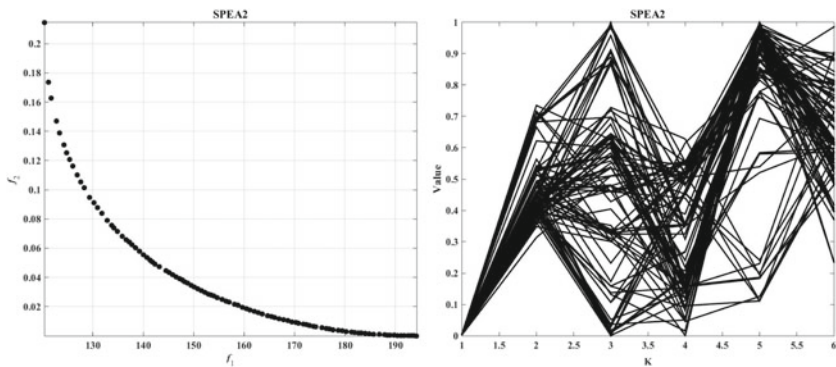
(e) SPEA2 Objective space two objective PIDs for Heat Exchanger

(f) SPEA2 Decision space

Fig. 3.11 Pareto approximates front and decision space (controller parameters) obtained from NSGA2, MOPSO, and SPEA2 algorithms for heat exchanger problem
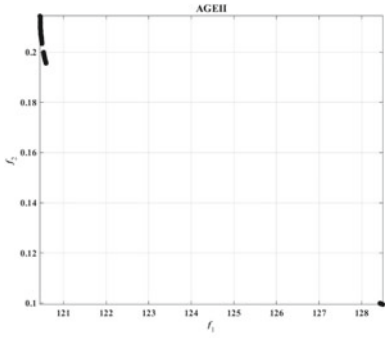
(a) AGE2 Objective space two objective PIDs for Heat Exchanger
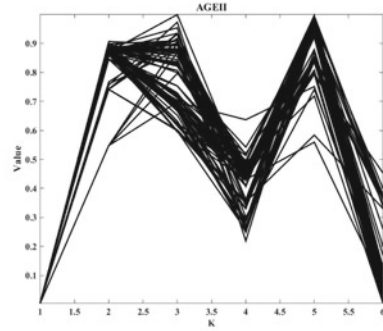
(b) AGE2 Decision space
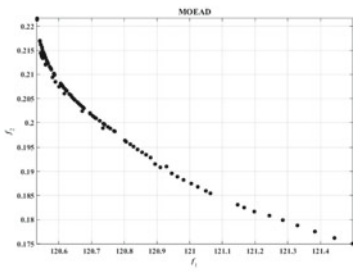
(c) MOEAD Objective space two objective PIDs for Heat Exchanger
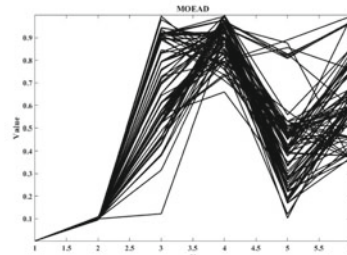
(d) MOEAD Decision space
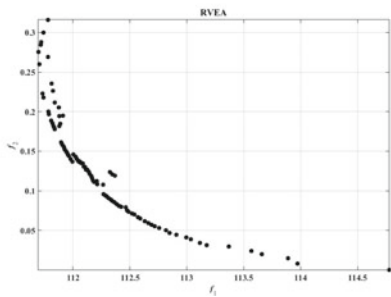
(e) RVEA Objective space two objective PIDs for Heat Exchanger
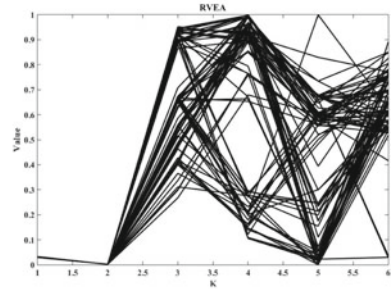
(f) RVEA Decision space

Fig. 3.12 Pareto approximates front and decision space (controller parameters) obtained from AGE2, MOEAD, and RVEA algorithms for heat exchanger problem
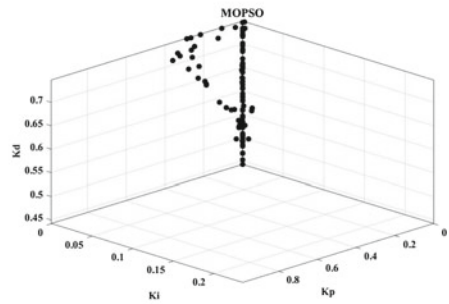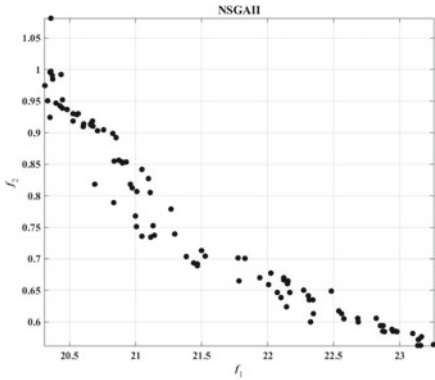
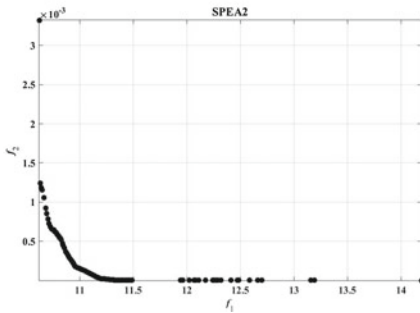(a) NSGA2 Objective space two objective PIs for Distillation Column
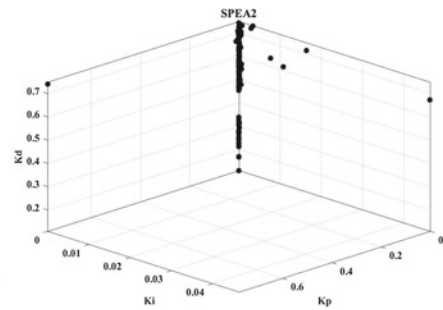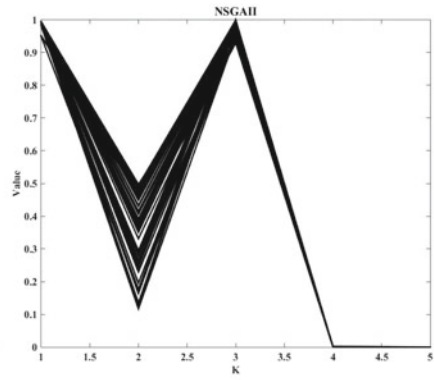
(b) NSGA2 Decision space
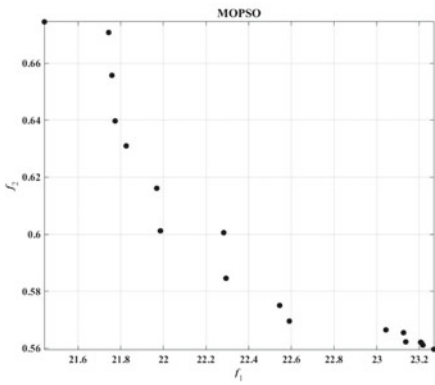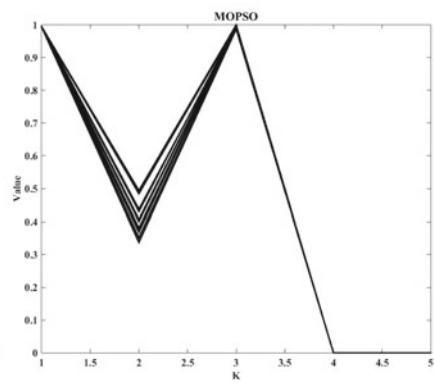
(c) MOPSO Objective space two objective PIs for Distillation Column

(d) MOPSO Decision space
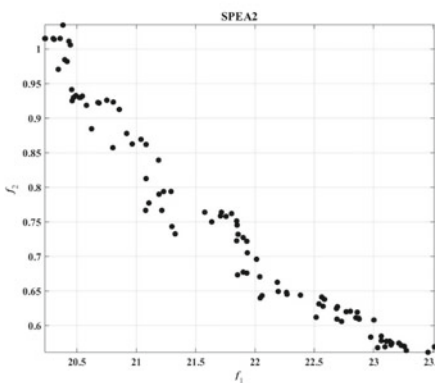
(e) SPEA2 Objective space two objective PIs for Distillation Column
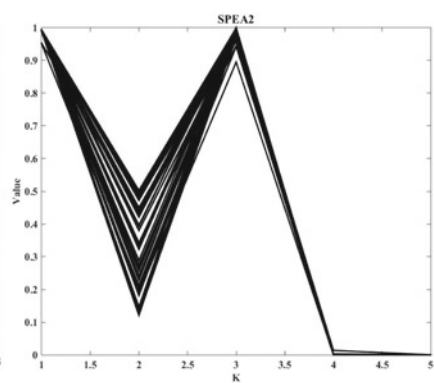
(f) SPEA2 Decision space

Fig. 3.13 Pareto approximates front and decision space (controller parameters) obtained from NSGA2, MOPSO, and SPEA2 algorithms for distillation column problem
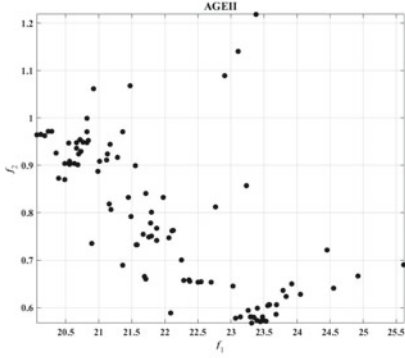
(a) AGE2 Objective space two objective PIs for
Distillation Column
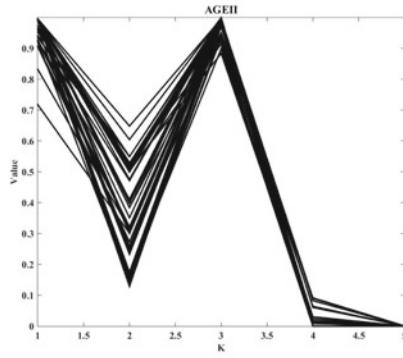
(b) AGE2 Decision space
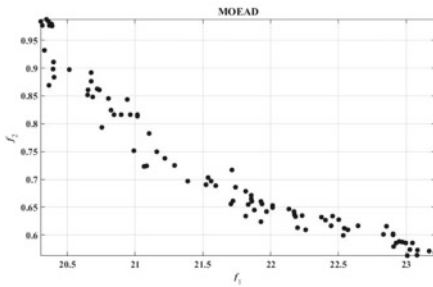
(c) MOEAD Objective space two objective PIs for
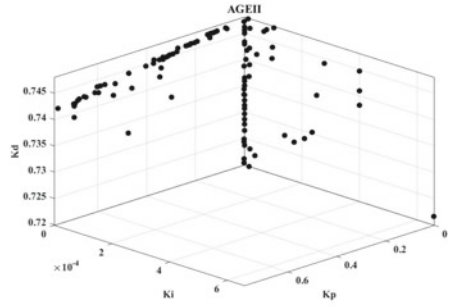Distillation Column
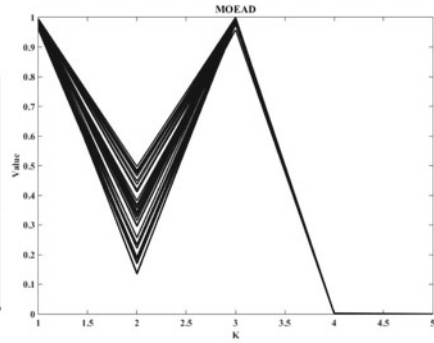
(d) MOEAD Decision space
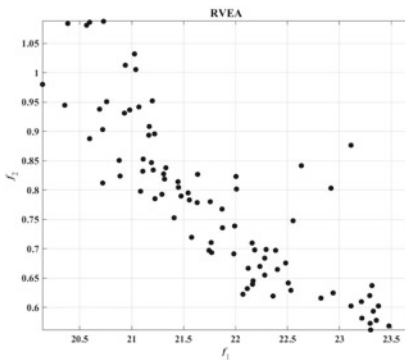
(e) RVEA Objective space two objective PIs for
Distillation Column
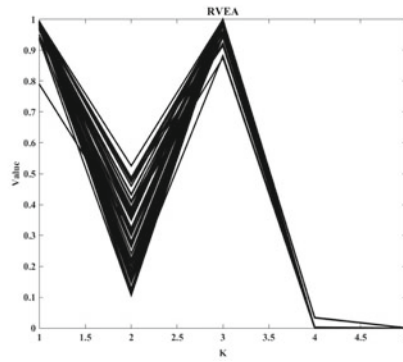
(f) RVEA Decision space

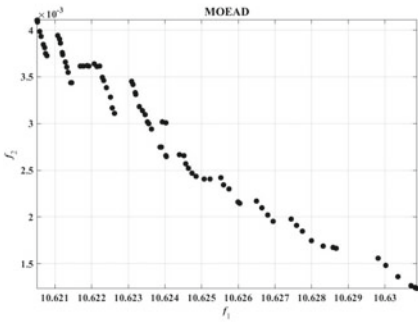**Fig. 3.14** Pareto approximates front and decision space (controller parameters) obtained from
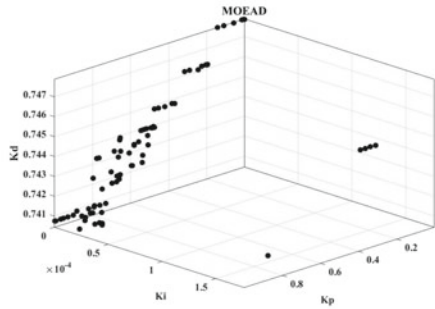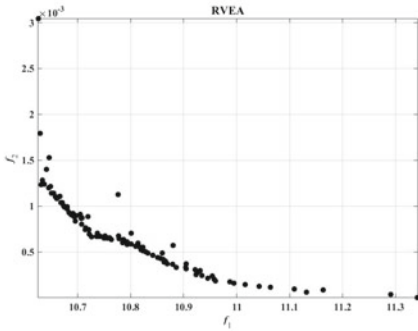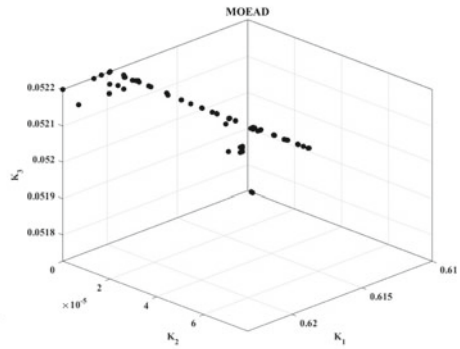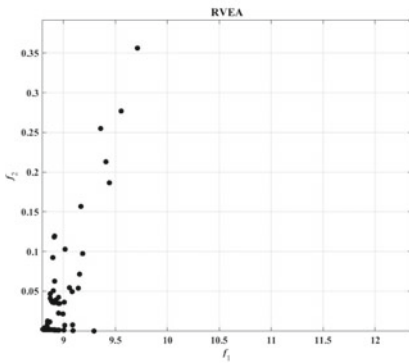AGE2, MOEAD, and RVEA algorithms for distillation column problem

3.5 Conclusion

In this study, the optimal parameters of the controller algorithm are obtained by using six different optimization algorithms. For this purpose, three real-life industrial devices are selected as test problems. The results are compared both graphically and numerically. From the results, it is observed that the best performance can be observed in MOEAD, when all of the problems are considered. Also, the multi-variable system is a hard problem, even if it has only a five-dimensional decision space. Therefore, as the future study, the controller parameter tuning algorithms for multi-variable systems will be investigated more efficiently, and new tools will be proposed to find the optimal controller parameters.

References

1. Jagatheesan, K., Anand, B.: Automatic generation control of three area hydro-thermal power systems with electric and mechanic governor. In: Computational Intelligence and Computing Research, pp. 18–20 (2014)
2. Laghari, J., Mokhlis, H., Bakar, A.B.H.A., Mohammad, H.: Comparative studies on load frequency control for isolated distribution network connected with mini hydro. In: Power Engineering and Optimization Conference, pp. 211–216 (2011)
3. Ziegler, J.G., Nichols, N.B.: Optimum settings for automatic controllers. Trans. ASME **64**, 759–768 (1942)
4. Cohen, G.H., Coon, G.A.: Theoretical consideration of retarded control. Trans. ASME **75**, 827–834 (1953)
5. Rivera, D.E., Morari, M., Skogestad, S.: Internal model control 4. PID controller design. Ind. Eng. Chem. Process Des. Dev. **25**, 252 (1986)
6. Ram, P., Jha, A.N.: Automatic generation control of interconnected hydro thermal system in regulated environment considering generation rate constraints. In: International Conference on Industrial electronics, Control and Robotics, pp. 148–159 (2010)
7. Khuntia, S.R., Panda, S.: A novel approach for automatic generation control of a multi-area power systems. In: Conference on Electrical and Computer Engineering, pp. 1182–1187 (2011)
8. Singh, M.K., Naresh, R., Gupta, D.K.: Optimal tuning of temporary droop governor of hydro power plant using genetic algorithm. In: International Conference on Energy Efficient technologies for Sustainability, pp. 1132–1137 (2013)
9. Nanda, J., Mishra, S., Mishra, P.G., Sajith, K.V.: A novel classical controller for automatic generation control in thermal and hydrothermal systems. In: International Conference on Power Electronics, Drives and Energy Systems, pp. 1–6 (2010)
10. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: 1995 IEEE International Conference on Neural Networks Proceedings, vols. 1–6, pp. 1942–1948 (1995)
11. Eberhart, R.C., Shi, Y.H.: Particle swarm optimization: Developments, applications and resources. In: Proceedings of the 2001 Congress on Evolutionary Computation, vols. 1 and 2, pp. 81–86 (2001)
12. Man, K.F., Tang, K.S., Kwong, S.: Genetic algorithms: concepts and applications. IEEE Trans. Ind. Electron. **43**, 519–534 (1996)
13. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**, 341–359 (1997)
14. Dangor, M., Dahunsi, O.A., Pedro, J.O., Ali, M.M.: Evolutionary algorithm-based PID controller tuning for nonlinear quarter-car electrohydraulic vehicle suspensions. In: Nonlinear Dynamics, vol. 78, pp. 2795–2810, Dec 2014

15. Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S.: GSA: a gravitational search algorithm. Inf. Sci. **179**, 2232–2248 (2009)
16. Tseng, C.-S., Chen, B.-S.: Multiobjective PID control design in uncertain robotic systems using neural network elimination scheme. IEEE Trans. Syst. Man Cybern. Part A: Syst. Hum. **31**(6), 632–644 (2001)
17. Takahashi, R.H.C., Peres, P.L.D., Ferreira, P.A.V.: Multiobjective H2/H1 guaranteed cost PID design. IEEE Control Syst. **17**(5), 37–47 (2002)
18. Tang, K.S., Man, K.F., Chen, G., Kwong, S.: An optimal fuzzy PID controller. IEEE Trans. Ind. Electron. **48**(4), 757–765 (2001)
19. Liua, G.P., Daley, S.: Optimal-tuning PID control for industrial systems. Control Eng. Pract. **9**(11), 1185–1194 (2001)
20. Ayalaa, H.V.H., Coelho, L.S.: Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator. Expert Syst. Appl. **49**(10), 8968–8974 (2012)
21. Panda, S.: Multi-objective PID controller tuning for a FACTS-based damping stabilizer using non-dominated sorting genetic algorithm-II. Int. J. Electr. Power Energy Syst. **33**(7), 1296–1308 (2011)
22. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2 Improving the strength Pareto evolutionary algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich (2001)
23. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Parallel Problem Solving from Nature PPSN VI (2000)
24. Coello Coello, C.A., Lechuga, M.S.: MOPSO: a proposal for multiple objective particle swarm optimization. In: CEC, pp. 1051–1056 (2002)
25. Coello Coello, C.A., Pulido, G.T., Lechuga, M.S.: Handling multiple objectives with particle swarm optimization. IEEE Trans. Evol. Comput. **8**(3), 256–279 (2004)
26. Rodrigez, C.E.R., Bideaux, C., Guillouet, S.E., Gorret, N., Roux, G., Jouve, C.M., Lara, C.A.A.: Multi-objective particle swarm optimization (MOPSO) of lipid accumulation in Fed-batch cultures. In: 24th Mediterranean Conference on Control and Automation, pp. 979–984 (2016)
27. Zhang, Q., Li, H.: MOEA/D: a multi-objective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. **11**(6), 712–731 (2007)
28. Wagner, M., Neumann, F.: A fast approximation-guided evolutionary multi-objective algorithm. In: Evolutionary Computation Group Technical Document, pp. 1–8 (2013)
29. Cheng, R., Jin, Y., Olhofer, M., Sendhof, B.: A reference vector guided evolutionary algorithm for many-objective optimization. IEEE Trans. Evol. Comput. **20**(5), 773–791 (2016)
30. Schott, J.R.: Fault tolerant design using single and multicriteria genetic algorithm optimization. M.S. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts (1995)

Chapter 4
Fuzzy and Neuro-fuzzy Control for Smart Structures

Georgios K. Tairidis and Georgios E. Stavroulakis

Abstract Classical control tools often encounter a number of limitations on the investigation of smart composite structures due to nonlinearities and/or other uncertainties. Especially in smart structures, which is the case here, a significant degree of uncertainty is involved due to several imperfections and/or errors of both the controller and the structure itself. For example, in structures with multiple layers, several failures may appear, such as delamination, debonding, fatigue, etc. The use of intelligent fuzzy and adaptive control which is based on neuro-fuzzy techniques can be very helpful in this direction. One may also consider using global optimization algorithms for the fine-tuning of the characteristics of the controllers to maximize their applicability, their efficiency, and their robustness. In other words, the controllers can be designed based on intuition and basic engineering principles, and then they can be subjected to optimization, e.g., to training/learning using artificial neural networks, in order to achieve certain properties.

4.1 Introduction

It is well known that vibration suppression on smart structures can be achieved using active control. It is also known that classical mathematical control tools usually provide satisfactory results for linear feedback laws under given assumptions. However, the design of nonlinear controllers based on fuzzy inference systems and/or artificial neural networks, or even hybrid neuro-fuzzy controllers can provide satisfactory results when the system is partially known.

The key advantage of classical control theories is the availability of strong mathematical tools for the design of the controllers. However, the fact that a linear feedback

G. K. Tairidis (✉) · G. E. Stavroulakis
School of Production Engineering and Management, Technical University of Crete,
Institute of Computational Mechanics and Optimization, 73100 Chania, Greece
e-mail: tairidis@gmail.com

G. E. Stavroulakis
e-mail: gestavr@dpem.tuc.gr

is adopted is considered to be a serious drawback. On the other hand, nonlinear control, e.g., fuzzy control, is more flexible and more suitable to handle nonlinearities. This is due to the fact that the feedback of such controllers is nonlinear and could be of varying intensity in different areas of operation. This is not only expected but desirable as well, as a nonlinear controller can serve different needs, e.g., slight or more extensive displacements, with the same initial settings. Specifically, for the fuzzy inference systems, the presence of verbal rules can systematize the experience of an advanced user of a system or a process and can be used for the construction of nonlinear controllers. The control output could be nonlinear and complicated.

Fuzzy control offers an efficient interface for the translation of the knowledge of one system into the nonlinearities which are necessary for the control. An introduction to fuzzy control is made in [1], where all the necessary tools for building robust fuzzy controllers are presented in detail. Moreover, an introductory survey of fuzzy control has been conducted in [2]. The theoretical framework as well as some industrial and other fuzzy applications is covered in the aforementioned references and in recent review articles like [3–6]. The most important features of a controller based on fuzzy logic are, among others, the ability to function under multiple objectives and the adaptiveness to different problems. Another significant advantage of fuzzy control is the robustness, while the major disadvantage seems to be the absence of a complete mathematical framework for the description and the study of such systems.

Fuzzy and neuro-fuzzy control systems are capable of solving hard problems. Several optimization techniques such as the adaptive neuro-fuzzy inference system (ANFIS) procedure which is based on neural networks, as well as global optimization tools, such as genetic algorithms, particle swarm optimization, differential evolution, etc., can be very useful in order to improve the characteristics of the control [7, 8]. In the present chapter, which is an extended version of [9], the main tools two control strategies, i.e., the fuzzy and the adaptive neuro-fuzzy, are presented in terms of the reduction of oscillations of smart structures.

4.2 Fuzzy Control

4.2.1 Fuzzy Logic

Fuzzy inference systems quantify linguistic logical rules by using the fuzzy theory on suitably fuzzified variables. Fuzzy inference systems can be used for the creation of rule-based control. The recurring systems can be used for the control of various processes, from many scientific fields, where the transfer of existing, and maybe empirical, knowledge is important, and smooth transition between control strategies plays a significant role. The whole idea is based on a quite modern science of reasoning which in turn is called fuzzy logic. Operators use common sense to solve complex problems. Fuzzy logic is a set of mathematical principles that is used to represent
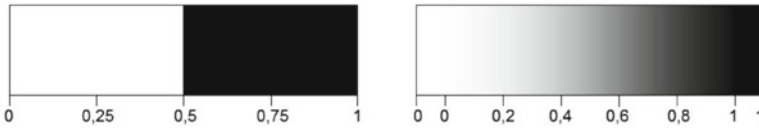
Fig. 4.1 Boolean logic (left) versus fuzzy logic (right)

experienced operators' knowledge to a computer system. In essence, fuzziness is not a part of the logic. In fact, a set of logical expressions is used to describe the fuzziness that exists in most systems. The representation is based on the membership level of the involved parameters that interact with each other through a set of verbal rules.

According to Boolean logic, concepts and variables are divided in a sharp manner. For instance, if we divide men in groups according to their height, setting the limit of 1.80 m as the minimum height for the tall ones, someone with 1.79 m height is considered short, while a man of 1.81 m height is considered tall.

Fuzzy logic is based on the principle that all parameters or concepts of a system do not have a single meaning, but instead they are subjected to ratings. Distance, speed, temperature, height, weight, service, etc., can be described using scales. For example, George is very tall. Service is below average and so on. In other words, fuzzy logic reflects how people think, avoiding generalization errors verbalizing variables and simulating the human sense. In that meaning, it does not face things in a black-and-white scale, but on the contrary, it includes many intermediate states (gray scale) see Fig. 4.1. This property leads to the construction of smart control systems.

The forefather of fuzzy logic was the probability theory, according to which each parameter could belong in a set with a percentage from 0 to 1. For example, we can say that if someone has a height of 1.81 m. is tall with probability 1, very tall with probability 0.8 and short with probability 0. However, the father of fuzzy logic, as we know it today, is L. Zadeh, who in 1965 in his article entitled "Fuzzy sets" introduced the concept of fuzzy sets using the mathematical tools of the probability theory, introducing a totally new logic which was based in verbal terms which he called "fuzzy logic" [10].

4.2.2 Membership Functions

The degree of fuzziness of a fuzzy set is defined by its membership functions. The representation of these functions can be done either numerically or graphically. The graphical representations include various forms, each with its own restrictions. The membership functions can have any parameterized form, either symmetric or asymmetric. The most popular forms include among others: triangular, trapezoidal, bell, Gaussian, sigmoid, and polynomial membership functions. A graphical representation of a membership function in comparison with a crisp set is shown in Fig. 4.2.
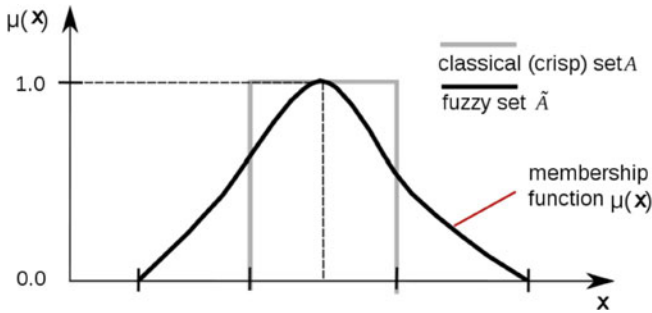
Fig. 4.2 A fuzzy membership function in comparison to a crisp set (https://commons.wikimedia.org/wiki/File:Fuzzy_crisp.svg)

4.2.3 Fuzzification and Defuzzification

Fuzzification is among the most important processes in the fuzzy theory. In particular, it is the process of converting an explicit numerical quantity into a fuzzy one, which is represented by the membership functions. The process is based on the recognition of the uncertainty which exists in explicit quantities. On practical applications, it is possible for errors to occur with a consequent reduction of data accuracy. This reduction of precision can also be represented by the membership functions. The definition or fine-tuning of the membership functions can be done either intuitively or by using algorithms and logical processes. The most popular methods include, among others intuition, inference, rank ordering, angular fuzzy sets, neural networks, genetic algorithms, inductive reasoning, etc.

With the term defuzzification is denoted the conversion process of the fuzzy outputs into explicit values. This process is necessary as the value of outputs must be accurate, especially when the fuzzy system is used as a controller, where the fuzzy outputs are not useful for further processing. For the defuzzification of fuzzy output functions, several methods can be used, such as the maximum membership principle, the centroid, the bisector, the middle or mean of maximum (MOM), the smallest of maximum (SOM), the largest of maximum (LOM), the center of sums, the center of largest area, etc. (see Fig. 4.3).

The choice of the appropriate defuzzification method is often a subjective process and depends on the data and/or the requirements of each problem. It is worth mentioning that two different methods can give completely different results. It is also possible the results of two or more methods to be identical. For example, if the final surface is triangular, the result of methods SOM, MOM, and LOM will be identical (the top of the triangle). In neuro-fuzzy controllers, the calculation of the final outputs is based on the computation of the weighted average of outputs as

$$z_0 = \frac{\sum_{i=1}^{n} a_i \cdot C_i}{\sum_{i=1}^{n} a_i} \tag{4.1}$$

Fig. 4.3 The most common
defuzzification methods
(http://www.mathworks.
com/help/examples/fuzzy_
featured/defuzzdm_04.png)
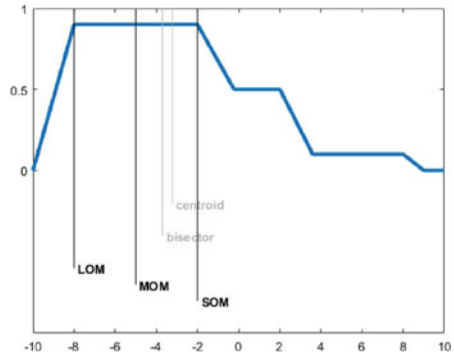
where a_i are the trigger points of membership functions and C_i are the individual values of the outputs.

It is worth mentioning that weighted average is not exactly a defuzzification method, in the context which this term is met in the Mamdani-type controllers, but it is usually called like that due to the fact that it is used for the calculation of the final outputs.

4.2.4   Fuzzy Inference Systems

Fuzzy inference systems or FIS are also known as fuzzy rule-based systems or fuzzy models. The rules of verbal variables can be formed by deterministic statements (e.g., velocity = high), condition statements (e.g., IF grade $\geqslant 8.5$ THEN excellent) or statements without condition (e.g., GO TO). The properties of the set of the rules are the fullness, consistency, continuity, and interaction. A fuzzy system is usually described with more than one rules. The process of summarizing the rules for obtaining an overall conclusion is called aggregation. If the individual rules are associated with the AND operator, the determination of the aggregation is done by the conjugation of the rules, by taking the intersection, while for the OR operator, the determination of the aggregation is done by the disjunction of the rules, calculating the union of the involved rules. The methods of conjugation and disjunction are also known as methods of minimum (min) and maximum (max), respectively.

The structure of a fuzzy inference system includes a set of IF-THEN rules, a set of membership functions, a decision-making unit, which is also called inference process, a fuzzification interface and a defuzzification interface. The operation of the inference system goes as follows. The explicit inputs are converted into fuzzy via fuzzification. Then the set of rules is drafted, which together with the data, forms the knowledge database. Subsequently, the decision is made by implication, and the fuzzy output arises. Finally, this value is defuzzified. The whole process is depicted schematically in Fig. 4.4.
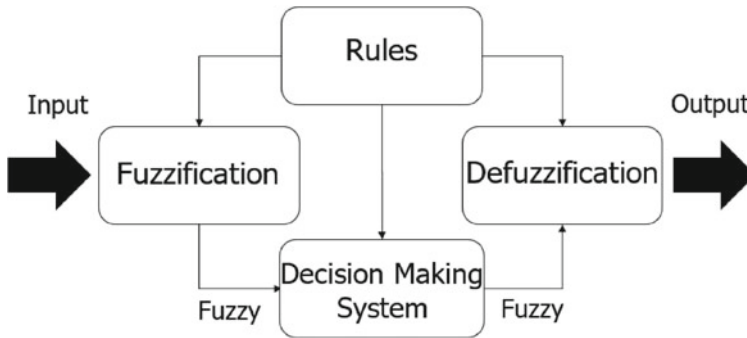
Fig. 4.4 The structure of a fuzzy inference system

4.2.5 Fuzzy Inference Methods

The two main methods of fuzzy inference are the Mamdani method and the Sugeno method. Other known methods are the Inference of Larsen and the Tsukamoto method. The Mamdani method, which is the most widespread, was introduced by Mamdani and Assilian in 1975 [11]. Ten years later Takagi and Sugeno introduced the Sugeno method, which is also known as Takagi–Sugeno method [2]. These two methods have several common characteristics.

Their main difference lies in the type of membership functions of their outputs. In Mamdani method, the membership functions are fuzzy sets. Instead, in the Sugeno method, the outputs are either linear functions or constant values. The main advantages of each of the two methods of fuzzy inference are summarized as follows.

The Mamdani method is an intuitive method, which is widely accepted, and it adapts well to real problems. It is a relatively simple method which works well even in complex models, without sacrificing accuracy. The basic steps of the implementation process of the Mamdani method are as follows:

1. Fuzzification of inputs using membership functions,
2. Definition of verbal rules of fuzzy system,
3. Evaluation of rules,
4. Calculation of system outputs, and
5. Defuzzification.

The characteristics (membership functions, rules, etc.) of a fuzzy controller can be optimized using global optimization methods inspired by nature, such as genetic algorithms [7], particle swarm optimization [8, 12], differential evolution [13], etc.

The Sugeno method is a computationally accurate method, which works very effectively in combination with linear techniques. It also works effectively in combination with optimization techniques. Moreover, the Sugeno method presents a guaranteed continuous output surface, and it is susceptible to mathematical analysis. Likewise to the Mamdani method, the steps for the Sugeno inference are as follows:

1. Fuzzification of input variables (clustering),
2. Determination of system rules,
3. Evaluation of system rules, and
4. Calculation of outputs.

In the Sugeno controllers, fuzziness of inputs and their categorization (clustering) in membership functions is similar to that followed by Mamdani-type controllers. The same applies to the rules governing the fuzzy system. However, the major advantage of Sugeno-type controllers is the fact that they can be trained using adaptive neuro-fuzzy inference (ANFIS) techniques, as it will be described in Section 4.4.

4.2.6 Fuzzy Control Applications

The most important features of a controller based on fuzzy logic are, among others, the ability to function under multiple objectives and the adaptiveness to different problems [1]. Another significant advantage of fuzzy control is the robustness. On the other hand, the major disadvantage seems to be the absence of a complete mathematical framework for the description of such systems [1, 2]. However, there is a plethora of applications of fuzzy control in literature.

A control paradigm for smart structures using fuzzy control is presented in [14]. In this study, fuzzy control techniques and smart piezoelectric materials are combined in order to form a robust and adaptive control system. The proposed model is capable of learning and thus of improving its performance over time in presence of uncertainties in the host plant, so as to be adaptive. The robustness is proved by changing the properties of the structure, i.e., by adding some extra masses to the host beam.

A modal control law of cantilever piezoelectric beams using fuzzy logic is presented in [15]. The inputs of the controller are the modal displacements and modal velocities of the modes to be controlled. The performance of the proposed method is proved for the first two modes of the host structure. The results obtained, indicate that the implementation of fuzzy controllers for active vibration control is not only feasible, but effective as well.

In [16] the vibration control of a smart plate is considered. The model consists of a thin elastic rectangular plate, while the controller is designed using fuzzy inference techniques. In this investigation, two different computational procedures are proposed. In the first approach, a local controller is considered, while in the second algorithm, the controller is distributed. From the numerical results, both methods seem to achieve the desired vibration suppression of the host plate.

Fuzzy schemes, able to deal with system nonlinearities, consist a suitable solution for the control of complex systems, like control based on magnetorheological damping [17], or for the smart and adaptable usage of classical controllers like the fuzzy-PID control of seismically excited structures reported in [18].

Fuzzy systems have been extensively used for control of air conditioners, energy systems, and consumer electronics (cameras, washing machines, etc.), see, among

others, recent publications like [19–25]. As an example of a recent application on complex systems, the control of a photovoltaic energy system connected to a desalination plant is treated in [26].

In [27], a methodology for the identification of damage in structures is proposed. In this investigation, fuzzy techniques combined with continuum damage mechanics are used to recognize both the location and the extent of the failure. The suggested methodology is a new approach which monitors static and dynamic responses and can be used for the damage detection of various civil structures. The numerical examples presented in this paper indicate that the performance of fuzzy systems is vigorous under conditions of noise or high uncertainty. See also the related material in [28] and the recent publication [29].

In [30] a fuzzy rule-based model for the prediction of delamination in drilling of glass fiber reinforced plastic composites has been introduced. The results of the analysis are verified with experimental ones, indicating the efficiency of the proposed method.

Finally, guidance, control, and fault identification in automotive and unmanned vehicles and robotic systems have been studied with fuzzy control techniques, see, among others, [31–34].

4.3 Artificial Neural Networks

Machine learning is a scientific field that includes adaptive methods, which in turn allow computers to be trained based on experience, examples, and proportionality. A core characteristic of these methods lies in the fact that learning abilities improve the performance of a machine learning system over time. An artificial neural network (ANN) is an approach of machine learning which attempts to simulate the function of the human central nervous system, i.e., of the biological neural networks. It is about a network of interconnected calculating nodes (artificial neurons) which are algorithms of computational intelligence.

4.3.1 What Is a Neural Network?

A biological neural network is a model of logical thinking which is based on the human brain. Our cerebrum consists of a 10 billion neural cells network and these cells have 60 trillion connections that are called synapses [35]. The cell is the structural element of a neural network. Each cell has a simple structure; however, the combination of a huge number of cells provides incredible computational power, allowing the rapid processing of stimuli by the human brain. In a neural network, besides the body (soma) of the cell, there is also the axon, the synapses, and the dendrites (see Fig. 4.5).
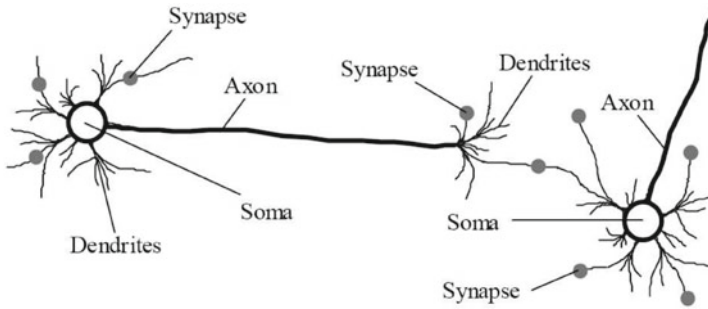
Fig. 4.5 Biological neural network

Any given neuron of a biological neural network is able to respond only to a certain subset of stimuli which are within its receptive field, i.e., its sensory space [36]. This key property of the networks is called tuning. This core characteristic of biological neural networks has inspired the design and formulation of artificial neural networks. Thus, the comparative advantage of an artificial neural network is the fact that it is also open to training; in other words, it has plasticity. This process, known as learning, trains the network in order to be capable of better solving several problems. This is done through a repeated process where the parameters of the network are self-adjusted. Once trained, the network receives inputs from the environment (stimuli) processes them and provides an output (decision), which is sent either to the environment or to the next neuron until the final processing is made. If the training is done properly, the network will be able to solve even problems for which it has not been trained for. This means that it should be able to produce outputs even for inputs that it ignores, which in turn is the objective. This property is called generalization.

The first artificial neural network was designed by McCulloch and Pitts in 1942 [37]. This was one of the first studies of computer systems that relied on the functioning of the human brain. In 1957, Rosenblatt introduced the concept of the neuron perceptron [38]. In 1986, Rumelhart, Hinton, and Williams suggested the backward propagation method of errors, known simply as back-propagation method [39].

Neural networks are used in a wide range of applications from different sciences. Some of these are systems control, pattern recognition, stock market control, equipment maintenance, various robotics applications, etc. The main advantages of such networks over other heuristic methods are among others, their ability to solve highly complex problems, their tolerance to the existence of noise, as well as the fact that they do not require previous knowledge of the model. This latter feature makes neural networks particularly useful in systems where experimental measurements exist, but there is a lack of information about the model. In this case, these experimental measurements can be used for training and control of the network.
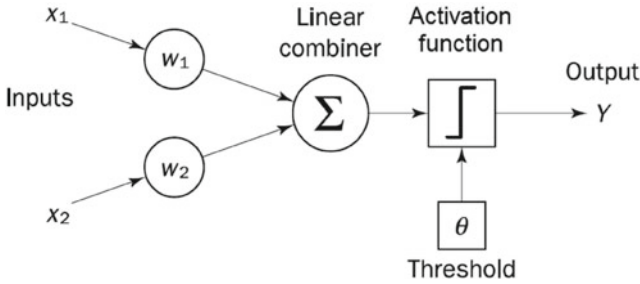
Fig. 4.6 A typical neuron

4.3.2 The Concept of the Artificial Neuron

Artificial neurons are the structural elements of artificial neural networks. In fact, they are nodes which take inputs and produce outputs. The inputs can be received from other neurons or directly from the environment. Similarly, the output is sent either to the external environment or is used as input to other network neurons. An artificial neuron consists of a set of inputs x_i, a set of synaptic weights w_i, a bias or threshold θ, an activation function f and the output of the neuron y or O as seen in Fig. 4.6.

Generally, there are three different types of neurons: the input neurons, the output neurons, and the hidden neurons. The role of the input neurons is to import the inputs of the network to the intermediate (hidden) neurons where the calculations take place. There, each input is multiplied by the corresponding synaptic weight. The sum of the products of these multiplications is being calculated and then inserted in the activation function, which is calculated at each node. The value of the function is the output of the hidden neuron for the given inputs. By nature, neurons located in the so-called hidden layers have the ability to "hide" their desired output. For this reason, no information from the interaction between inputs and outputs can be extracted. Moreover, there is no obvious way to predict the desired values of the outputs of hidden neurons.

4.3.3 Calculation of Outputs

A neuron with n inputs $x_1, \ldots, x_n$ where every synapsis has a synaptic weight $w_1, \ldots, w_n$ is considered. The input x_0 is always 1 and the synaptic weight w_0 is the threshold for the activation of the neuron. The output y_k of every neuron is calculated by the following equation:

$$y_k = f\left(\sum_{i=0}^{n} x_{ki} \cdot w_{ki}\right) \tag{4.2}$$

Step function	Sign function	Sigmoid function	Linear function

$$Y^{step} = \begin{cases} 1, & \text{if } X \geq 0 \\ 0, & \text{if } X < 0 \end{cases} \quad Y^{sign} = \begin{cases} +1, & \text{if } X \geq 0 \\ -1, & \text{if } X < 0 \end{cases} \quad Y^{sigmoid} = \frac{1}{1 + e^{-X}} \quad Y^{linear} = X$$
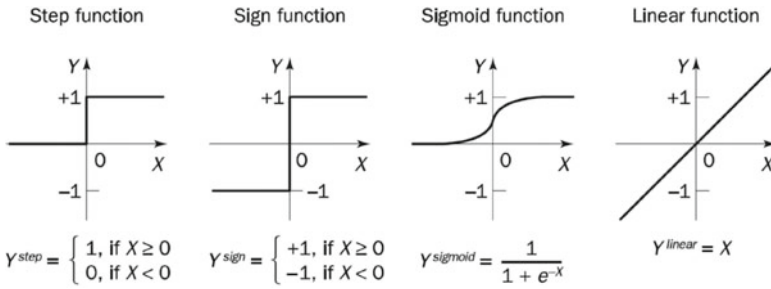
Fig. 4.7 Graphic representation of the most common activation functions

where x_{ki} and w_{ki} are the i-th input and weight of the neuron k, respectively, n is the number of neurons and f is the activation function.

A neuron is positively activated when

$$\sum_{i=0}^{n} w_i \cdot x_i \geqslant w_0 \tag{4.3}$$

The weight $w_0 = \theta$ is called threshold. If θ is equal to zero, then the input is disregarded. The activation of the neurons of an artificial neural network can be done using several activation functions, e.g., a step function, a sign function, a linear function, a nonlinear function, etc. (see Fig. 4.7).

4.3.4 Training of Neural Networks

The objective of the training procedure is to find the appropriate weights of the synapses, for which the network will be able to produce the desired values of the outputs for given inputs. There are several ways of learning, and thus training of artificial neural networks, which can be classified into two major categories: supervised learning and unsupervised learning.

Supervised learning is the process that combines the existence of an external trainer and the total amount of the available information for the model. This category includes the back-propagation method of errors, the perceptron method, the stochastic learning, etc. In supervised learning, the supervisor has to decide on a number of issues such as the examples (experimental data) that will be used for training, the progress of the training process, the termination of the training, etc. The back-propagation method of errors is a common training method for multilayer neural networks, that is, networks with many layers. Quite often, it is used in combination with other methods such as the steepest descent algorithm. The method calculates the derivative of the error function considering all the network weights.

Fig. 4.8 Back-propagation process
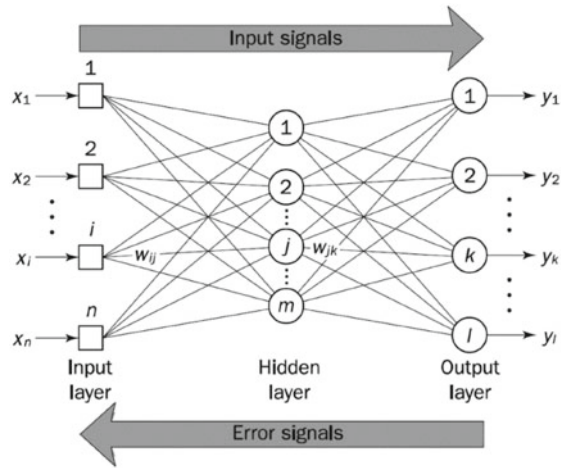
The calculated derivative is fed to the optimization method, which in turn uses it to update the weights in order to minimize the error. This method requires knowledge of the desired outputs for every input, in order to calculate the derivative of the error. Another requirement of this method is the differentiability of the activation function which is used by the neurons. The whole process is depicted in Fig. 4.8.

The unsupervised training of an artificial neural network tends to follow the neurobiological organization of the human brain. This type of training is very fast and therefore can be used in real-time applications. The training is done by self-organized algorithms, which train the network based only on local information without requiring the existence of a supervisor. During the training, the neural network receives an amount of training data, finds similarities or important features of these data and learns to classify them into categories. Some of the most popular methods of unsupervised training are, among others, the Hebb training algorithm, the competitive learning (Kohonen networks), etc. However, the presentation of these methods does not fall within the scope of this chapter.

It should be noted here that supervised learning solves a complex, possibly global optimization problem for the determination of the optimal parameters (weights) of the neural network, using the existing input–output values of the examples.

4.4 Adaptive Neuro-fuzzy Inference Systems (ANFIS)

Simple fuzzy systems are very popular in several scientific fields, such as control, robotics, etc. The basic structure of these systems relies on membership functions for inputs and outputs, as well as a set of verbal rules in order to establish the decision-making system as described in Section 4.2. Membership functions should be chosen

by experience or arbitrarily and the structure of the rules should be predetermined and based on user's interpretation of the model.

Such systems are very efficient when applied to control. However, there are some critical limitations, such as the absence of a systematic framework or method of transforming human experience into a set of if-then rules, and/or the lack of an intact methodology for the fine-tuning of the parameters of fuzzy controllers, and especially of membership functions. Moreover, the application of fuzzy inference techniques to systems, for which a set of input/output data already exists, is quite common. Structural control is only one example. In many industrial applications, the model is more or less known, or a collection of measurements could be easily obtained.

It is rather often that when the control mechanism is built, the designer or the engineer cannot decide the form and the other characteristics of the membership functions, or the coherence of the rules just considering the available data. In this case, adaptive fuzzy systems can solve the problem.

4.4.1 What Is ANFIS?

ANFIS is one of the most popular adaptive fuzzy systems, if not the most popular. A thorough study of the adaptivity of fuzzy systems, especially in control, as well as their stability properties, can be found in classical monographies as "Adaptive fuzzy systems and control: Design and stability analysis" [40].

The forerunner of adaptive network-based fuzzy systems was a fuzzy system which was modeled using generalized neural networks (GNN) and a Kalman filter algorithm to minimize the squared error [41]. In this approach, a fuzzy inference system with parameters that could be updated was built. From the simulation results was extracted that the proposed fuzzy system is able to fine-tune its parameters (e.g., the membership functions of inputs), as well as to incorporate prior knowledge about the original system.

The architecture of ANFIS is based on a fuzzy inference system which in turn is implemented inside the framework of adaptive neural networks and introduced by Jyh-Shing R. Jang at University of California in 1993 [42]. ANFIS consists of fuzzy rules which, in contrast to classical fuzzy systems, are local mappings instead of global ones [43]. These mappings facilitate the minimal disturbance principle, which states that the adaptation should not only reduce the output error for the current training pattern, but also minimize disturbance to response already learned [44]. This is particularly important if an online learning process is considered. Comparisons with neural network approaches can be found in [42].

The process which should be followed in order to create a fuzzy inference system is usually called fuzzy modeling. On the other hand, neuro-fuzzy modeling refers to the way of applying various learning techniques developed in the neural network literature to fuzzy inference systems. Back-propagation neural networks are mostly used for the identification of the parameters of an adaptive fuzzy inference system.

The learning procedure could be hybrid, i.e., the proposed control model can construct an input–output mapping based on both human knowledge, just like in fuzzy systems, and appropriate input/output data pairs. However, even if human expertise is unavailable, it is still possible to set up the initial parameters intuitively and generate the fuzzy rules using a learning process in order to approximate the desired performance. This means that, rather than choosing the parameters of the controller (membership functions, rules, etc.) arbitrarily, an automated process can provide tailor-made membership functions for the fuzzy variables (inputs and outputs) based on the available system's data. Moreover, a set of rules or other parameters of the control can also be considered and the most important; the controller can be trained in order to be robust, i.e., capable of functioning under different conditions.

In general, fuzzy control is one of the most successful applications of the fuzzy theory. However, due to the adaptive capability that ANFIS technique provides, fuzzy control becomes even more powerful, to the extent that it could be able to replace neural networks in control systems.

4.4.2   What Is the ANFIS Routine of MATLAB?

In this case, the acronym ANFIS is derived from adaptive neuro-fuzzy inference system, and it can be used in MATLAB as part of the fuzzy logic toolbox. Namely, it is a training routine for Sugeno-type fuzzy inference systems. With ANFIS, a fuzzy inference system can be constructed, just using a given input/output data set. The parameters of the system can be adjusted using either a back-propagation algorithm alone or in combination with an algorithm based on the least-squares method. This tuning allows fuzzy systems to learn from the same data they are modeling. The learning method works similar to that of neural networks [45].

The modeling approach is similar to many system identification techniques. First, a parameterized model is considered and then a set of data for training is collected and applied. The parameters of the fuzzy system will be adjusted automatically using these data until an error criterion is met. It is essential that the training data (data for learning) are carefully chosen [46]. This means that in general, as for simple models, the more the data the better the approximation; however, for noisy systems or when the collected data are not representative of the system, model validation might be helpful. This validation can be achieved using another data set (data for testing). In general, model validation is the process by which inputs, on which the system was not trained, are presented to the trained model, to check the accuracy of the prediction. This is necessary because after a certain point in the training process, the model may overfit the training data. The testing data also allow the designer to check the generalization capability (robustness) of the resulting fuzzy inference system. An ANFIS model structure in MATLAB is depicted in Fig. 4.9.
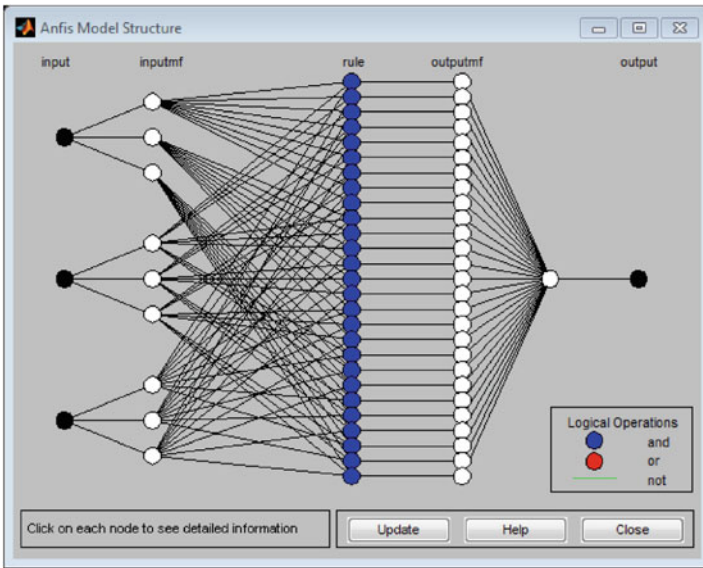
Fig. 4.9   An ANFIS model structure in MATLAB

4.4.3   Training of Adaptive Neuro-fuzzy Inference Systems Through MATLAB

The first step of the learning procedure is the collection of a set of training examples with the desired input/output data of the system to be modeled. These data must be an array with the data arranged as column vectors, and the output data in the last column. The data for training could be loaded either from a file or from the MATLAB workspace.

4.4.3.1   Initialization of the System Parameters

The initial fuzzy inference system parameters could be parameterized manually or, if there is no preference or experience on how they should look like, ANFIS can initialize the parameters automatically. This means that the model structure can be loaded either by a previously saved Sugeno-type fuzzy inference system structure or generated by choosing one of the following partitioning techniques: grid partition, which generates a single-output Sugeno-type fuzzy inference system by using grid partitioning on the data; or subtractive clustering, which generates an initial model for ANFIS training by first applying subtractive clustering on the data.

A typical grid partition in a two-dimensional input space is one of the most common options when designing a fuzzy controller, especially if the desired number

of the clusters is known. This method usually considers only certain of the parameters of the controller, such as the input variables. This partition strategy works perfect for a small amount of membership functions for each input [43]. However, when a moderately large number of inputs exist, grid partition method encounters serious problems. For instance, a fuzzy model with 10 inputs and two membership functions on each input would result in $2^{10} = 1024$ fuzzy if-then rules, which is prohibitively large. This problem, usually referred to as the curse of dimensionality and can be alleviated by other partition strategies such as tree partition and scatter partition which both overcome the problem of the exponential increase in the number of fuzzy rules by covering only a subset of the input space, which in turn needs to be carefully selected by the designer.

Subtractive clustering, on the other hand, is the suitable option if the designer of the controller does not have a clear idea of how many clusters there should be at each input for a given set of data [47]. Moreover, it is a fast, one-pass algorithm for estimating the number of clusters and the cluster centers in a set of data. These estimates can be used to initialize iterative optimization-based clustering methods and model identification methods like ANFIS.

4.4.3.2   The Training Process Through ANFIS

After loading the training data and generating the initial FIS structure, the training process can be continued. As mentioned above, there are two optimization methods available; the back-propagation of errors and the hybrid method, which is a combination of least-squares error method (LSE) and back-propagation. Both methods are used in order to train the membership function parameters to emulate the available training data.

The back-propagation method is a gradient descent method, while the least-squares method is a standard approach in regression analysis, which is used for the computation of an approximate solution in overdetermined systems, i.e., systems of equations with more equations than unknowns. The term least squares suggests that the overall solution minimizes the sum of the squares of the computational errors of every single equation.

The hybrid method is based on back-propagation for the calculation of the parameters associated with the input membership functions, and on the least-squares method for the estimation of the parameters related to the output membership functions. It is found that the use of least-squares method for the calculation of outputs of each local mapping is very important, as according to some researchers the learning time without using LSE would be ten times longer [42].

The number of training epochs and the error tolerance are the stopping criteria for training and are both set by the designer of the model. The training process stops whenever one of the above criteria is met, i.e., when the maximum epoch number is reached, or the training error goal is achieved. If the impact of the training error
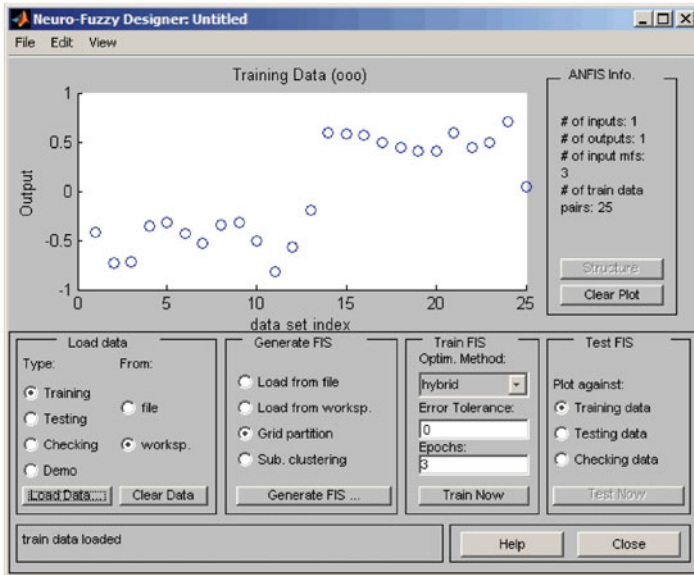
Fig. 4.10 An example of training through ANFIS of MATLAB

to the results is unknown, error tolerance should be set to zero. An example of a training process through the ANFIS routine of MATLAB is depicted in Fig. 4.10.

To make the training process more accurate, the results of each iteration are used as the initial conditions for the next epoch. The training error, which occurs in the output, decreases, at least locally, throughout the learning process. This means that, the more the initial membership functions approach the optimal ones, the easier it will be for the training algorithm to converge. Human knowledge or expertise about the target system can be of great assistance in setting up these initial parameters of the fuzzy inference system.

4.5 Fuzzy and Neuro-fuzzy Controllers

In the present investigation, two different intelligent controllers are built. More specifically, a Mamdani-type fuzzy controller, as well as a Sugeno-type neuro-fuzzy controller are implemented and tested. The basic characteristics of these controllers are shown below.
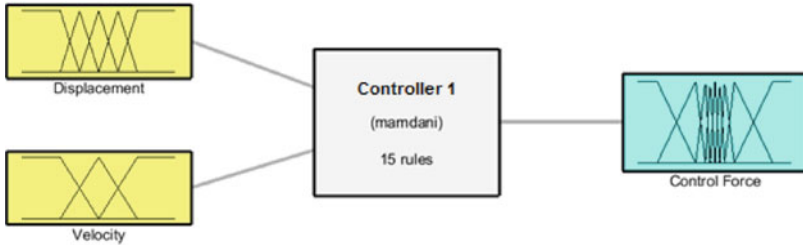
Fig. 4.11 Fuzzy inference system of Mamdani fuzzy controller
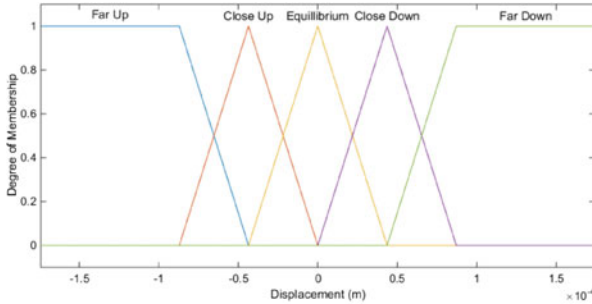
Fig. 4.12 Membership function of displacement (input 1)
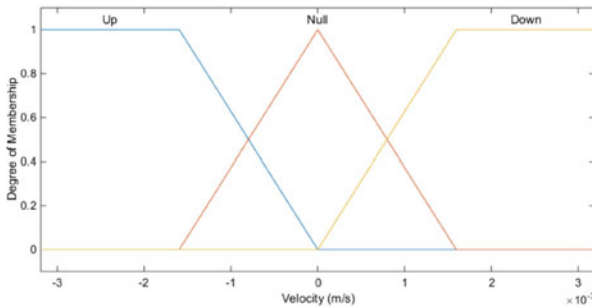
Fig. 4.13 Membership function of velocity (input 2)

4.5.1 Fuzzy Controller

The fuzzy inference system (see Fig. 4.11) is developed within MATLAB using the fuzzy toolbox [48]. The control scheme is a Mamdani-type controller of two inputs (displacement and velocity) and one output (control force).

The membership functions of displacement, velocity, and control force have triangular and trapezoidal form, as shown in Fig. 4.12, Fig. 4.13 and Fig. 4.14, respectively.
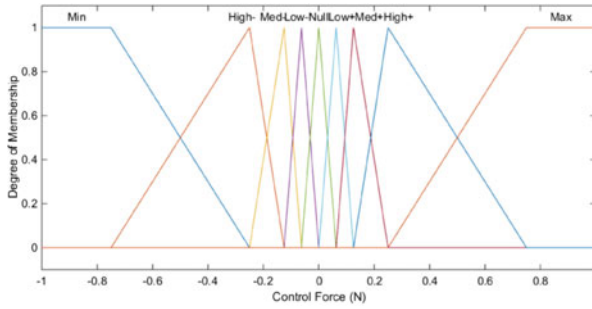
Fig. 4.14 Membership function of control force (output)

Table 4.1 Fuzzy rules of the fuzzy controller (e.g., if displacement is far up and velocity is up then the control force is max)

Displacement velocity	Far up	Close up	Equilibrium	Close down	Far down
Up	Max	Med+	Low+	Nul	Low-
Nul	Med+	Low+	Null	Low-	Med-
Down	High+	Null	Low-	Med-	Min

Fig. 4.15 Graphic representation of the rules of fuzzy controller
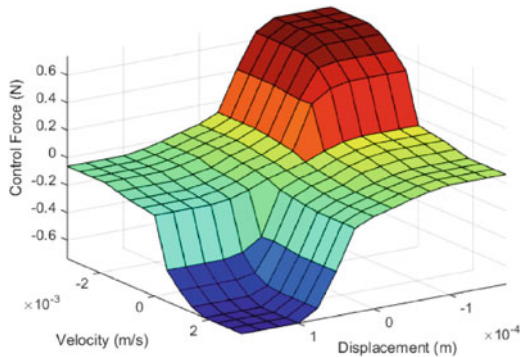
The inference system involves a decision-making system which is based on the combination of the membership functions with the use of logical operations (see Fig. 4.4). Namely, the decision (output) is computed through a set of if-then rules; thus, the recurring system is a rule-based system.

For the implementation of the present fuzzy controller, a set of 15 rules is used as shown in Table 4.1. All rules have weights equal to unity and are connected using the AND operator. The graphic representation of the linguistic rules is shown in Fig. 4.15. The implication method is set to minimum and the aggregation method is set to maximum. As for the defuzzification, several methods can be used, as described in Subsection 4.2.3.
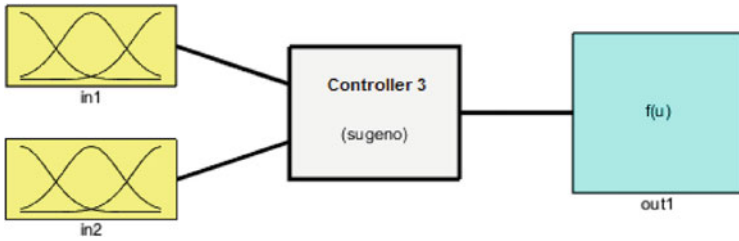
Fig. 4.16 Fuzzy inference system of Sugeno neuro-fuzzy controller
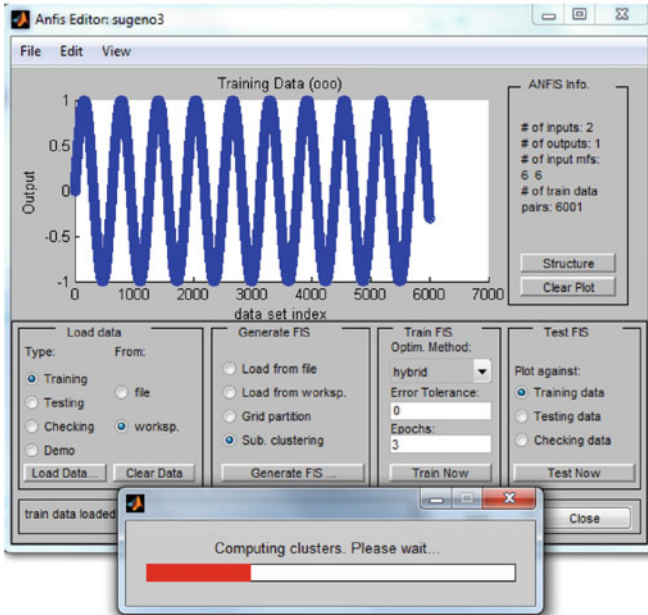
Fig. 4.17 Computing of the clusters of the Sugeno controller using subtractive clustering

4.5.2 Neuro-fuzzy Controller

In contrast to Mamdani-type fuzzy controllers, the adaptive neuro-fuzzy control schemes that are presented here are based on Sugeno-type controllers. For this controller, again, the system takes two inputs (displacement and velocity) and returns one output (control force), thus is a multiple-inputs-single-output (MISO) control device [49]. The overview of the Sugeno controller is shown in Fig. 4.16.

Optimization of the controllers is achieved via a training process within the adaptive neuro-fuzzy inference system (ANFIS) package of MATLAB (see Fig. 4.17).

As mentioned in Subsection 4.2.5, only Sugeno-type controllers are eligible for training with ANFIS. The characteristics of the fuzzy system in this case are tuned
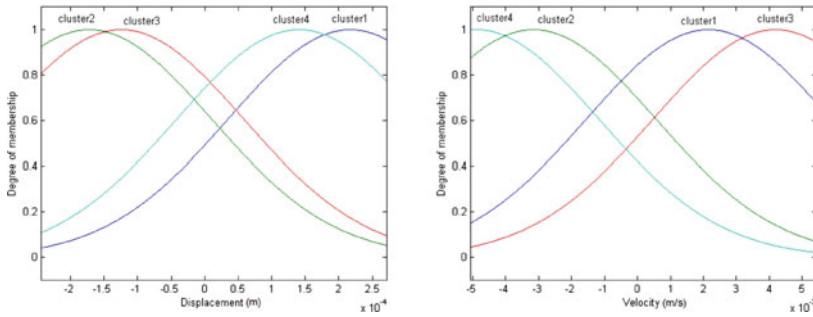
Fig. 4.18 Membership functions of displacement and velocity

Table 4.2 Fuzzy rules of the ANFIS controller (e.g., if displacement is in cluster 1 and velocity is in cluster 1 then the control force is out1)

Displacement velocity	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Cluster 1	Out1	–	–	–
Cluster 2	–	Out2	–	–
Cluster 3	–	–	Out3	–
Cluster 4	–	–	–	Out4

using artificial neural networks. More specifically, a well-chosen set of training data is used to adjust the system parameters. For the compilation of these data, the model is first simulated without any control mechanism attached, to collect the necessary vibration data. Subsequently, these data are used for the training of the Sugeno controller.

Once loading the data to the ANFIS editor of MATLAB, the initial fuzzy inference system can be generated either with grid partitioning on the data, if the form and the number of the membership functions is known, or, otherwise, if there is no such information, subtractive clustering may be used (see Subsection 4.4.3). The procedure of the computation of the clusters of the inputs of this controller via the subtractive clustering method through the ANFIS editor is shown in Fig. 4.17. The resulting form of the membership functions (clusters) of the inputs, i.e., the displacement and the velocity, after the initialization process is shown in Fig. 4.18. Namely, four clusters of Gaussian form for each input occurred from the subtractive clustering process. The verbal rules which describe the emerging system are given in Table 4.2.

The structure of the rules in ANFIS is given in Fig. 4.19, and a visualization of them is depicted in Fig. 4.20.

The output variable, i.e., the control force, takes constant values within the range [−1 1]. The "and" method has been set to product, while the "or" method has been set to probabilistic or. The implication and the aggregation method have been set to min and max, respectively. As defuzzification method, the weighted average method is

Fig. 4.19 The structure of
rules in ANFIS
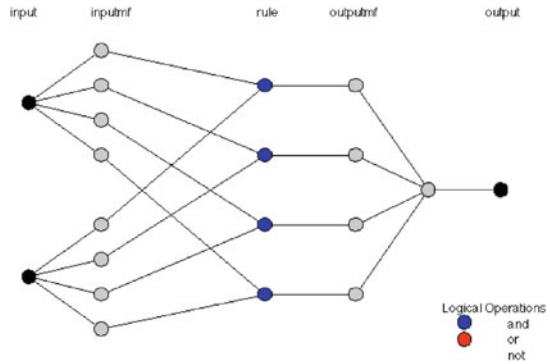
Fig. 4.20 Graphic
representation of the rules of
neuro-fuzzy controller
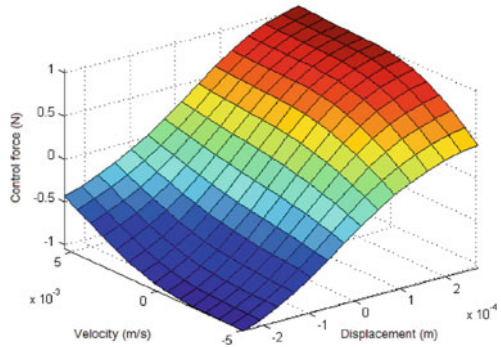
chosen. It is worth mentioning that the characteristics of this controller (membership
functions, rules, etc.) came through a training process, unlike the ones of fuzzy
controllers which have been set based on experience.

4.6 Numerical Examples

4.6.1 Fuzzy Control of a Cantilever Beam

The problem of the vibration suppression of a cantilever smart beam is first consid-
ered. The beam has a total length of 0.8 m and a square cross section with dimensions
0.02×0.02 m. The elasticity modulus is equal to 73×10^9 N/m^2. The mass density
is equal to 2,700 Kg/m^3. Excitation of sinusoidal form was applied concentrated at
the free end of the cantilever and is given as

$$P = P_0 \cdot sin(\omega \cdot t) \tag{4.4}$$

Table 4.3 Material properties of the aluminum and the piezoelectric material

Property	Aluminum	Piezoelectric
E_1 (GPa)	73	69
ρ (kg/m^3)	2700	7600
d_{31} (m/V)	–	210×10^{-12}
g_{31} (Vm/N)	–	11.5×10^{-3}

The amplitude P_0 of the excitation is equal to 1 N, while the frequency is 20 rad/s. The structure has been discretized using the finite element method. The control is collocated, i.e., it is applied to the same exact point where the measurement is obtained.

The material properties of the beam and of the piezoelectric material are given in detail in Table 4.3 (see also [48, 50, 51]).

As mentioned above, the controller takes as input the displacement and the velocity of the free end and returns the control force. The results are compared with the ones obtained by classical control and namely by a linear quadratic regulator (LQR), where the whole state of the dynamical system is assumed to be known. Finally, the weights Q and R of the regulator have been chosen to be diagonal matrices with appropriate dimensions according to literature:

$$q = 1 \quad \text{and} \quad r = 0.00001 \tag{4.5}$$

while the matrices Q and R are given as

$$Q = q \times I \quad \text{and} \quad R = r \times I \tag{4.6}$$

where I is the identity matrix.

The results for the vibration suppression in terms of displacement and velocity for the two kinds of control (both LQR and fuzzy control) are presented in Fig. 4.21. With blue color (dashed line) are the results prior to control, while the red color (solid line) represents the results after the application of the controllers.

From the results, one clearly observes that with suitably defined parameters for the fuzzy control system, very effective results in terms of displacement could be obtained. In fact, the results are comparable to the ones of LQR control even if they were obtained by using much fewer inputs. Nevertheless, the reduction of the vibrations in terms of velocity is not so effective.
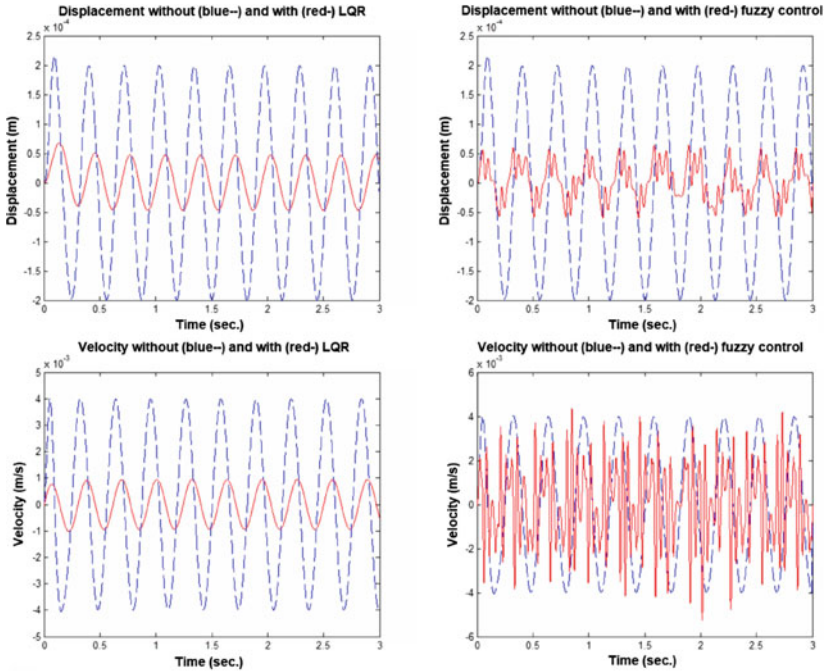
Fig. 4.21 Numerical results of fuzzy controller for the displacement and velocity compared to the ones of LQR control

4.6.2 Adaptive Neuro-fuzzy Control of a Cantilever Beam

In this case, the application of an optimized neuro-fuzzy controller for the suppression of the vibrations of a cantilever smart beam model with piezoelectric materials is presented. For the needs of the investigation, the adaptive Sugeno-type neuro-fuzzy controller, which was described above, is used.

The elastic part of the composite beam is made of aluminum material. The cantilever smart beam has a cross-sectional area which equals to 0.004 m^2, length 0.8 m, width 0.02 m, and height 0.005 m. The material properties of the elastic structure, as well as the ones of the piezoelectric material, are given in detail in Table 4.3 (see also [48, 50, 51]).

The structure is discretized using the finite element method, while the external force is assumed to be of sinusoidal form, and it is applied at the free end of the cantilever beam. The loading has the same form as given in Eq. (4.4). Again, the amplitude P_0 of the excitation is equal to 1 N, while the frequency is 20 rad/s. The adaptive Sugeno-type neuro-fuzzy controller is used for the reduction of the vibrations which are caused by the external loading. The model is first simulated without control in order to collect the necessary data for the training of the controller. Subsequently, the controller is trained by using the ANFIS procedure as described
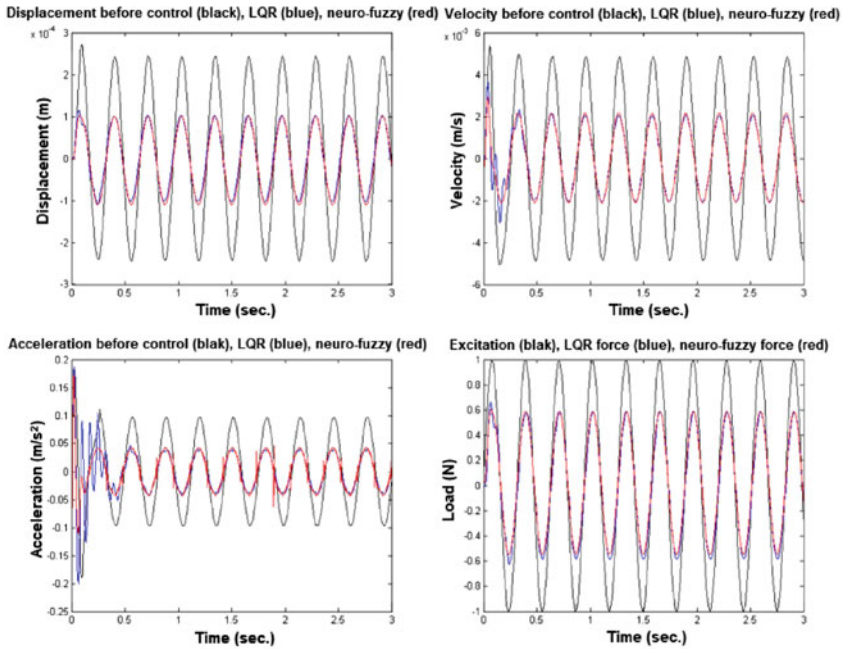
Fig. 4.22 Numerical results of neuro-fuzzy controller for the displacement, velocity, and acceleration, along with the forces

above and the trained controller is finally used to suppress the vibrations of the beam. The parameters of LQR control are given similar to the above example by Eqs. (4.5) and (4.6).

The numerical results of the trained adaptive neuro-fuzzy Sugeno controller are compared to the ones of a classical LQR controller (see Fig. 4.22).

The comparison shows that the results in terms of velocity and acceleration are quite impressive. For more information, one can see [49]. Vibration suppression for sinusoidal-type loadings on smart structures such as beams and plates can be achieved with Mamdani fuzzy controller without fine-tuning of the involved parameters [48]. The results are satisfactory for displacements, while velocities and accelerations are not acceptable. A comparison with a fuzzy controller [48] indicates that the vibrations are significantly lower after the application of the neuro-fuzzy controller, than the ones after the use of the simple fuzzy controller (see Fig. 4.23). One can say that the results of the Sugeno controller are quite comparable to the ones obtained by LQR control.

Another comparison is available in [52]. In this case, a ramp-type loading was chosen in order to show that Sugeno-type neuro-fuzzy controllers are not only smoother and more effective, but more stable as well, in comparison to simple fuzzy controllers. The results of this comparison are shown in Fig. 4.24. One remarkable achievement of neuro-fuzzy control is the adaptivity to different circumstances, other than the
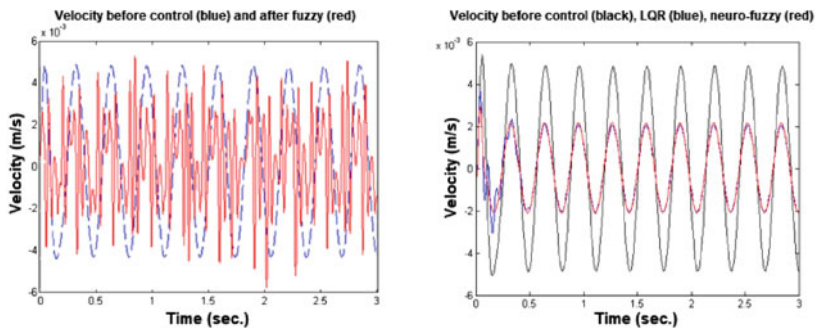
Fig. 4.23 Comparison of the velocity obtained by a fuzzy and a neuro-fuzzy controller
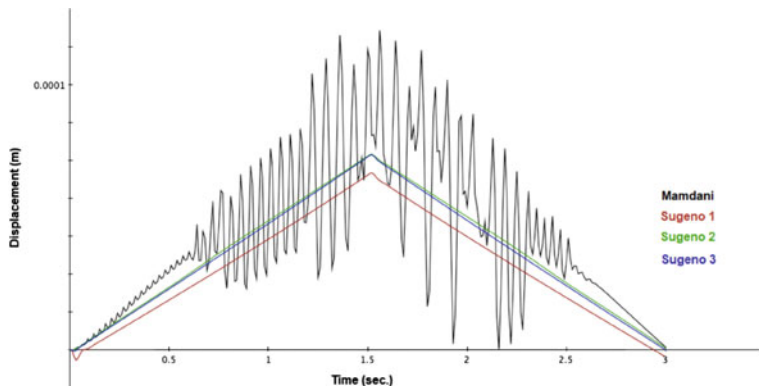
Fig. 4.24 Comparison of fuzzy and neuro-fuzzy controllers for a ramp-type loading

ones that it is trained for. For example, in this latter case, one Sugeno controller has been trained with a completely different type of loading, namely a sinusoidal one, and despite that, the control remains functional.

The same structure has been also tested with combinations of two local collocated fuzzy controllers, i.e., one near the free end and the second near the middle span of the cantilever beam. The numerical results validated the existence of the so-called smoothing effect, which is highly desired for prevention of fatigue phenomena of the involved materials and/or control devices.

4.7 Conclusions

Fuzzy control is quite efficient and let say smooth in terms of displacement and some times even of velocity reduction. However, the results in terms of acceleration are a burden for the material and the whole structure in general. These problems can be

addressed using optimization for the fine-tuning of the parameters of the control. The case here was the use of adaptive neuro-fuzzy inference control (ANFIS) techniques.

From the presented numerical results, one can conclude that adaptive neuro-fuzzy techniques can be useful for the development of smooth and robust controllers, capable of working in unsteady and varying environments. A systematic approach of adaptive fuzzy control is the hybrid neuro-fuzzy Sugeno controller trained by the ANFIS procedure, where the optimization of the system parameters is achieved via a training process of a suitable neural network. Another impressive conclusion is that adaptive neuro-fuzzy controllers can provide satisfactory vibration suppression, equivalent to classic control, such as the LQR, and the most important, without the need of knowledge of the full state-space of the problem or any other information. The only information which is needed is a set of appropriate data for the learning procedure, i.e., to train the neuro-fuzzy controller. It should be noted here once again that other optimization methods such as genetic algorithms [7], differential evolution [13], particle swarm optimization [8], etc., can be also used for the optimization of the control parameters with remarkable results. Instead of ANFIS, our team has also investigated the capabilities of direct optimization on the performance of a fuzzy control system on a whole trajectory with similarly promising results [12]. In fact, both approaches lead to smooth fuzzy controllers, which reduces damages due to fatigue in real-life applications.

Of course, a further investigation and/or the use of other adaptive neurocomputing control methods may help to better understand the proposed control schemes.

References

1. Driankov, D., Hellendoorn, H., Reinfrak, M.: An introduction to fuzzy control, 2nd edn. Springer, Munchen (1996)
2. Sugeno, M.: An introductory survey of fuzzy control. Inf. Sci. **36**, 59–83 (1985)
3. Precur, R.-E., Hellendoorn, H.: A survey on industrial applications of fuzzy control. Comput. Ind. **62**, 213–226 (2011)
4. Azadegan, A., Porobic, L., Ghazinoory, S., Samouei, P., Kheirkhah, A.-S.: Fuzzy logic in manufacturing: a review of literature and a specialized application. Int. J. Prod. Econ. **132**, 258–270 (2011)
5. Lu, P., Chen, S., Zheng, Y.: Artificial intelligence in civil engineering. Math. Probl. Eng. **2012**, Article ID 145974, 22 pp (2012)
6. Kar, S., Das, S., Ghosh, P.K.: Applications of neuro fuzzy systems: a brief review and future outline. Appl. Soft Comput. **15**, 243–259 (2014)
7. Tairidis, G., Foutsitzi, G., Koutsianitis, P., Stavroulakis, G.E.: Fine tunning of a fuzzy controller for vibration suppression of smart plates using genetic algorithms. Adv. Eng. Softw. **101**, 123–135 (2016)
8. Tairidis, G.K., Foutsitzi, G., Koutsianitis, P., Stavroulakis, G.E.: Fine tuning of fuzzy controllers for vibration suppression of smart plates using particle swarm optimization. In: 8th GRACM International Congress on Computational Mechanics Proceedings, Volos, 12–15 July (2015)
9. Tairidis, G: Optimal design of smart structures with intelligent control. Ph.D. thesis, Technical University of Crete, Greece.
10. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)

11. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. Int. J. Man. Mach. Stud. **7**, 1–13 (1975)
12. Marinaki, M., Marinakis, Y., Stavroulakis, G.E.: Fuzzy control optimized by a multi-objective particle swarm optimization algorithm for vibration suppression of smart structures. Struct. Multidisc. Optim. **43**, 29–42 (2011)
13. Marinaki, M., Marinakis, Y., Stavroulakis, G.E.: Fuzzy control optimized by a multi-objective differential evolution algorithm for vibration suppression of smart structures. Comput. Struct. **147**, 126–137 (2015)
14. Mayhan, P., Washington, G.: Fuzzy model reference learning control: a new control paradigm for smart structures. Smart Mater. Struct. **7**, 874–884 (1998)
15. Sharma, M., Singh, S.P., Sachdeva, B.L.: Fuzzy logic based modal space control of a cantilevered beam instrumented with piezoelectric patches. Smart Mater. Struct. **14**, 1017–1024 (2005)
16. Muradova, A.D., Stavroulakis, G.E.: Fuzzy vibration control of a smart plate. Int. J. Comput. Meth. Eng. Sci. Mech. **14**, 212–220 (2013)
17. Ding, J., Sun, X., Zhang, L., Xie, J.: Optimization of fuzzy control for magnetorheological damping structures. Shock. Vib. **2017**, Article ID 4341025, 14 pp (2017)
18. Baygi, S.M.H., Karsaz, A., Elahi, A.: A hybrid optimal PID-fuzzy control design for seismic excited structural system against earthquake: a salp swarm algorithm. In: 6th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), pp. 220–225, Kerman (2018)
19. Singh, J., Singh, N., Sharma J.K.: Fuzzy modeling and control of HVAC systems. A review. J. Sci. Ind. Res. **65**, 470–476 (2006)
20. Dounis, A.I., Caraiscos C.: Advanced control systems engineering for energy and comfort management in a building environment. A review. Renew. Sustain. Energy Rev. **13**, 1246–1261 (2009)
21. Bascetta, L., Rocco, P., Zanchettin, A.M., Magnani, G.: Velocity control of a washing machine: a mechatronic approach. Mechatronics **22**, 778–787 (2012)
22. Shaikh, P.-H., Nor, N.B.M., Nallagownden, P., Elamvazuthi, I., Ibrahim, T.: A review on optimized control systems for building energy and comfort management of smart sustainable buildings. Renew. Sustain. Energy Rev. **34**, 409–429 (2014)
23. Suganthi, L., Iniyan, S., Samuel, A.A.: Applications of fuzzy logic in renewable energy systems. A review. Renew. Sustain. Energy Rev. **48**, 585–607 (2015)
24. Haruki, T., Kikuchi, K.: Video camera system using fuzzy logic. IEEE Trans. Consum. Electron. **38**, 624–634 (1992)
25. Kyriakarakos, G., Dounis, A.I., Arvanitis, K.G., Papadakis, G.: A fuzzy logic energy management system for polygeneration microgrids. Renew. Energy **41**, 315–327 (2012)
26. Kyriakarakos, G., Dounis, A.I., Arvanitis, K.G., Papadakis, G.: Design of a fuzzy cognitive maps variable-load energy management system for autonomous PV-reverse osmosis desalination systems: a simulation survey. Appl. Energy **187**, 575–584 (2017)
27. Sawyer, J.P., Rao, S.S.: Structural damage detection and identification using fuzzy logic. AIAA J. **38**, 2328–2335 (2000)
28. Stavroulakis, G.E.: Inverse and crack identification problem in engineering mechanics. Kluwer Academic Publishers-Springer, Dordrecht, Boston, London (2000)
29. Jena, P.K., Thatoi, D.N., Parhi, D.R.: Dynamically self-adaptive fuzzy PSO technique for smart diagnosis of transverse crack. Appl. Artif. Intell. **29**, 211–232 (2015)
30. Latha, B., Senthilkumar, V.S.: Fuzzy rule based modeling of drilling parameters for delamination in drilling GFRP composites. J. Reinf. Plast. Compos. **28**, 951–964 (2009)
31. Ivanov, V.: A review of fuzzy methods in automotive engineering applications. Eur. Transp. Res. Rev. **7**, 29 (10 pp) (2015)
32. Chen, C.K., Dao, T.K.: Speed-adaptive roll-angle-tracking control of an unmanned bicycle using fuzzy logic. Veh. Syst. Dyn. **48**, 133–147 (2010)
33. Gupta, S.G., Ghonge, M.M., Jawandhiya, P.M.: Review of unmanned aircraft system (UAS). Int. J. Adv. Res. Comput. Eng. Technol. **2**, 1646–1658 (2013)

34. Kosari, A., Jahanshahi, H., Razavi, S.A.: An optimal fuzzy PID control approach for docking maneuver of two spacecraft: orientational motion. Eng. Sci. Technol. Int. J. **20**, 293–309 (2017)
35. Ivancevic, V.G., Ivancevic, T.T.: Brain and classical neural networks. In: Quantum Neural Computation. Intelligent Systems, Control and Automation: Science and Engineering, vol. 40. Springer, Dordrecht (2010)
36. Lodish, H., Berk, A., Zipursky, S.L., Matsudaira, P., Baltimore, D., Darnell, J.: Molecular cell biology, 4th edn. W. H. Freeman, New York (2000)
37. McCulloch, W.S., Pitts, W.H.: A logical calculus of ideas immanent in nervous activity. Bull. Math. Biophys. **5**, 115–133 (1942)
38. Rosenblatt, F.: The perceptron—a perceiving and recognizing automaton. Cornell Aeronautical Laboratory (1957)
39. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **32**, 533–536 (1986)
40. Wang, L.X.: Adaptive fuzzy systems and control: design and stability analysis. Prentice Hall, Upper Saddle River (1994)
41. Jang, J.-S.R.: Fuzzy modeling using generalized neural networks and Kalman filter algorithm. In: Ninth National Conference on Artificial Intelligence (1991)
42. Jang, J.-S.R.: ANFIS: adaptive-network-based fuzzy inference systems. IEEE Trans. Syst. Man Cybern. **23**, 665–685 (1993)
43. Jang, J.-S.R., Sun, C.-T.: Neuro-fuzzy modeling and control. Proc. IEEE **83**, 378–406 (1995)
44. Widrow, B., Lehr, M.A.: 30 years of adaptive neural networks: perceptron, madline, and back-propagation. Proc. IEEE **7**, 1415–1442 (1990)
45. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-fuzzy and soft computing; a computational approach to learning and machine intelligence. Prentice Hall, Upper Saddle River (1997)
46. Jang, J.S.R.: Input selection for ANFIS learning. In: Proceedings of IEEE 5th International Fuzzy Systems, New Orleans, LA, vol. 2, pp. 1493–1499 (1996)
47. Chiu, S.: Fuzzy model identification based on cluster estimation. J. Intell. Fuzzy Syst. **2**, 267–278 (1994)
48. Tairidis, G.K., Stavroulakis, G.E., Marinova D.G., Zacharenakis E.C.: Classical and soft robust active control of smart beams. In: Papadrakaikis, M., Charmpis, D.C., Tsompanakis, Y., Lagaros, N.D. (eds.) Computational Structural Dynamics and Earthquake Engineering, pp. 165–177. CRC Press, London (2009)
49. Tairidis, G.K., Papachristou, I., Katagas, M., Stavroulakis, G.E.: Neuro—fuzzy control of smart structures. In: 10th HSTAM International Congress on Mechanics Proceedings, Chania, 25–27 May (2013)
50. Foutsitzi, G., Marinova, D., Hadjigeorgiou, E., Stavroulakis, G. E.: Finite element modelling of optimally controlled smart beams. In: 28th Summer School: Applications of Mathematics in Engineering and Economics. Sozopol, Bulgaria (2002)
51. Stavroulakis, G.E., Foutsitzi, G., Hadjigeorgiou, V., Marinova, D.G., Baniotopoulos, C.C.: Design and robust optimal control of smart beams with application on vibrations suppression. Adv. Eng. Softw. **36**, 806–813 (2005)
52. Stavroulakis, G., Papachristou, I., Salonikidis, S., Papalaios, I., Tairidis G.: Neurofuzzy control for smart structures. In: Tsompanakis, Y., Topping, B.H.V (eds.) Soft Computing Methods for Civil and Structural Engineering, pp. 149–172, Saxe-Coburg, Stirlingshire, UK (2011)

Chapter 5
Computational Intelligence in the Desalination Industry

Pedro Cabrera and José A. Carta

Abstract Numerous studies have been undertaken since the start of the 1990s—when various authors began to propose the use of artificial intelligence in the field of water desalination—on the employment of computational intelligence (CI) systems in this technological field. The main goal of the proposals put forward has been to tackle the high degree of complexity involved in the different processes that can be found in the desalination industry. The wide variety of topics suggested as potential candidates for the application of CI in desalination processes include, among others, alarm processing and fault detection, control systems, operational optimization applications, load forecasting and security assessment. Although desalination plants have traditionally been powered by energy supplied by the burning of fossil fuels, there is a growing trend today, for various reasons, to use renewable energy sources to directly power these plants. This has added new challenges to the management of desalination processes as the temporal variability of renewable energy sources makes the decision-making processes more complicated. In turn, this means that a multivariable approach is required to ensure optimal desalination plant operation by maximizing the exploitability of the variable renewable resource. This chapter presents a review of how CI systems have been used to date in the desalination industry. A special mention is given to new developments which use CI systems to help overcome newly emerging challenges related to the increasing usage of renewable energy sources in the powering of desalination processes.

P. Cabrera (✉) · J. A. Carta
Department of Mechanical Engineering, University of Las Palmas de Gran Canaria,
Campus de Tafira s/n, 35017 Las Palmas de Gran Canaria, Canary Islands, Spain
e-mail: pedro.cabrerasantana@ulpgc.es

J. A. Carta
e-mail: jose.carta@ulpgc.es

5.1 Introduction

The incorporation of technologies for 'water desalination'—the conventionally accepted expression to define processes whose goal is to remove dissolved solids such as salts and minerals from water—has experienced exponential growth in the last few decades in many parts of the world [1–3]. The primary reason for the expanding use of water desalination technologies has been the serious freshwater crisis affecting many parts of the globe, especially in arid and semi-arid regions [4–6]. This crisis has several causes including, among others, changes to global weather patterns, an ever-increasing freshwater demand due to rising populations as well as a rising per capita water demand, environmental pollution due to the deterioration of existing aquifer resources and an agricultural and industrial expansion that has brought with it a high demand for fresh water.

Numerous water sources are used to feed the desalination technologies (brackish water, river water, seawater, etc.). However, the worldwide growth trend in terms of the installed capacity of seawater desalination technologies is far higher than that of the installed capacity of technologies used to desalinate other water sources. The installed capacities of the former are currently far superior to those of technologies which use other water sources [7].

There is a wide range of potentially usable desalination technologies [8, 9], but the ones presently in use can be broadly classified into two main types: those based on thermal processes and membrane separation-based technologies.

In thermal desalination processes, the water source is heated to produce steam which is subsequently condensed to obtain water of low salinity. The most commonly used thermal desalination technologies are multistage flash distillation (MSF) and multi-effect distillation (MED) methods. In 2011, MSF represented 23% of worldwide installed desalination capacity, and MED just 8% [7].

In membrane-based seawater desalination technologies, semipermeable membranes are used to separate solids from the water source. The most popular membrane-based technology is the process known as reverse osmosis (RO), in which the product water is separated from the dissolved solids of the feed water by pressure. Use of this technology has begun to rise sharply throughout the world, with the possible exception of regions in the Middle East. In 2011, RO technology was responsible for 63% of worldwide installed capacity [7]. Opting for this technology has been favoured by advances made in RO membranes and energy recovery devices which have enabled considerable reductions in specific energy consumption (kWh/m^3 of product water) [10, 11]. These reductions have in turn led to a decrease in freshwater production costs, which are closely related to energy costs [12]. The reasons why RO technologies, despite their increased market share in Middle East countries, have not prevailed over thermal desalination technologies which have far higher specific energy consumption [12] are related, on the one hand, to the fact that the installations in these countries are traditionally used to both produce water and generate energy and, on the other hand, to the low cost of fossil fuels in this region.

While the basic operating methods of the various desalination technologies are relatively simple [7, 12], the diverse range of tasks involved in the real-time operation of desalination installations creates a certain degree of complexity. For this reason, since the early 1990s when proposals first began to be made to use computational intelligence (CI) in the field of water desalination [13], numerous studies have been undertaken on how to tackle tasks such as alarm processing and fault detection, control systems, operational optimization applications, load forecasting and security assessment. Though these studies have considered thermal desalination processes like MSF [14–17] and other membrane-based processes like electrodialysis (ED) [18] and vacuum membrane distillation (VMD) processes [19] (an emerging technology that combines a thermal phase change with use of a membrane), the vast majority have centred on RO technology. This fact, along with the growing worldwide importance of RO water desalination because of lower energy requirements and higher water recovery rates compared to thermal systems, is one of the reasons why RO has been chosen for the purposes of the present chapter as representative of the desalination industry. Another reason for selecting RO technology is that proposals for the use of renewable energy sources to power desalination plants have centred on proposed or implemented systems which have been overwhelmingly RO-based [20]. Using renewable energies to power RO desalination plants, especially when these are combined with renewable generation technologies, means new challenges have to be faced in the operational and control management of renewable energy-powered desalination systems due to the temporal variability and inherent uncertainty of renewable energy sources. These new challenges involve the combined performing of multiple tasks, including system components management, control, forecasting, etc., that can be efficiently tackled through the use of CI.

This chapter is divided into five sections. The first section describes the basic problem of water scarcity and the principal solutions that have been proposed to date. The second section describes the basic theoretical concepts which allow an understanding of the operation of the most commonly used type of desalination plant (reverse osmosis desalination plant). The third section presents a review of how CI systems have been used to date in the desalination industry. In the fourth section, a special mention is given to recent developments which use CI systems to help overcome new challenges that have arisen as a result of the increased participation of renewable energy sources in powering desalination processes. The fifth and final section explains the most important conclusions of the chapter.

5.2 Basic Concepts of Reverse Osmosis Technology

Some basic notions are provided in this section of the technologies used in the desalination of seawater with reverse osmosis (SWRO) to enable the reader to better understand the studies and proposals made for the use of artificial intelligence (AI) in the desalination industry. In this context, a brief description of the RO process is first offered. This is followed by an also brief description of the basic configuration of an

RO desalination system, including some observations about the operating parameters. Any reader who would like to obtain a more in-depth and detailed knowledge of RO desalination technology is encouraged to consult the following publications [7, 12, 21].

5.2.1 Basic Notions on the Reverse Osmosis Process

When water solutions of low and high salinity are separated by a semipermeable membrane, the more diluted solution will pass through the membrane and dilute the more concentrated solution until the solutions on the two sides of the membrane reach the same concentration, Fig. 5.1. This natural phenomenon is known as the osmosis process. The hydraulic pressure which causes the flow of water from the side of low salinity to the side of high salinity is called the osmotic pressure, π, and is a function of the difference in ion concentration between the two sides of the membrane and of the temperature [7, 12].

If a pressure $p > \pi$ is applied to the more concentrated solution, the net operating pressure $\Delta P = p - \pi - p_l$ (where p_l are pressure losses due to the hydraulic resistance that is produced to the flow of water across the membrane) will make the water of that solution pass through the membrane in the opposite direction to that of the natural osmosis process, resulting in the RO process. The result is the separation of fresh water from the high salinity concentration, obtaining a more dilute solution on one side of the membrane (the membrane is semipermeable and is unable to prevent part of the dissolved solids in the more concentrated water from crossing it) and a more concentrated solution on the other, Fig. 5.1.
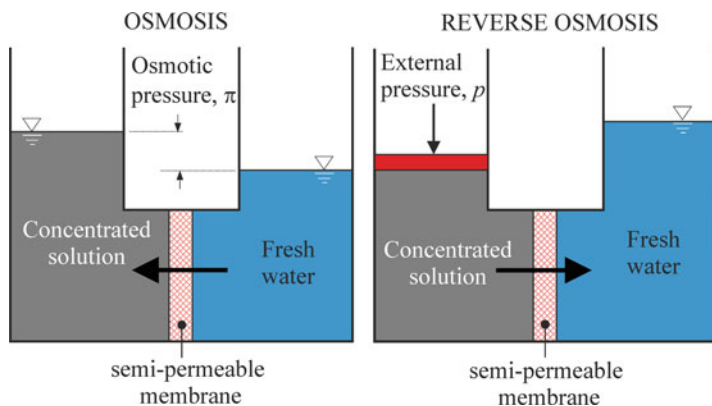
Fig. 5.1 Osmosis and reverse osmosis process

5.2.2 Components of a Typical SWRO Plant

The daily production capacities of an SWRO plant range from below 50 m³/day in smaller units to as high as 300,000 m³/day [12]. While the configurations of SWRO systems are highly diverse [7, 12], a typical SWRO plant will generally contain the following components, Fig. 5.2.

- A pump station whose function is to pump seawater from the intakes to the inlets of the high-pressure pumps responsible for feeding it into the SWRO membrane system. The water intakes vary in type (open intakes, wells, etc.) and can be situated offshore or onshore. The pumping system can be equipped with variable frequency drives (VFDs) which are used to regulate the flow rate of the pumped water.
- A physico-chemical seawater pre-treatment system, which aims to protect membranes against deposits of encrusting salts, fouling with sand, organic matter, colloidal matter and bacterial or chemical attack. For this, filter systems are used (for the removal of solids, organics and microorganisms), and the chemical products that are required (flocculants, coagulants, scale inhibitors, oxidants, etc.) are fed by metering pumps into the seawater.
- One or various SWRO systems, each one comprising:

  - A high-pressure feed pump (centrifugal pump or reciprocating pump), which raises the water pressure of the water that has passed through the filtering system from between 2 and 6 bar to between 55 and 70 bar, overcoming the osmotic pressure and the resistance of the hydraulic circuit. The electric generators that drive these pumps can be coupled to VFD that enable regulation of the flow rate of the water that is fed into the SWRO membrane system.
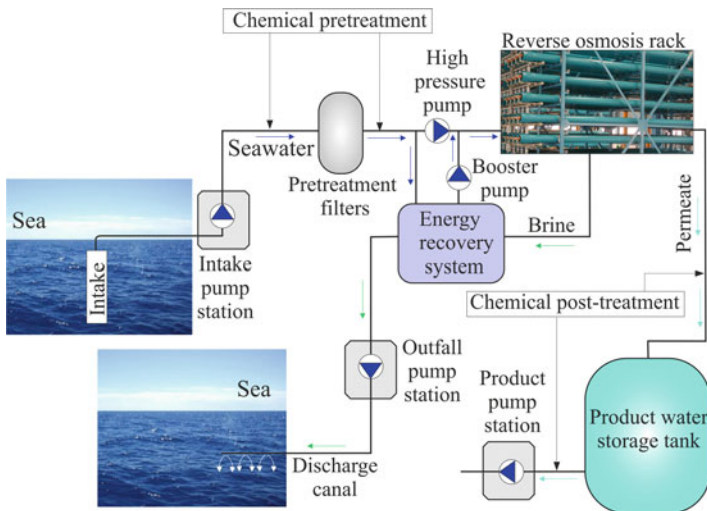
Fig. 5.2 Schematic representation of a seawater desalination plant

  – SWRO desalination membrane elements (generally of the spiral–wound polyamide type), which, normally numbering six or seven and having the same or differing productivity and salt rejection characteristics, are connected in series in the interior of a pressure vessel. One or multiple (hydraulically connected in parallel) pressure vessels are situated on support structures commonly known as racks. Each SWRO system can be configured with a single rack (single-pass SWRO system) or with two racks situated in series (two-pass SWRO system), with the result that the permeate water is treated in two 'passes'. In the case of a two-pass SWRO system, a booster pump is also installed between the two racks to increase the pressure of the permeate. To extract a higher flow of fresh water from the feed water (i.e. to increase the system recovery rate), the SWRO racks can be configured in two stages. In such a system, the reject flow of a rack becomes the feed flow of the next rack.
  It should be noted that an SWRO system can also be configured using multiple passes and stages.
  – An energy recovery system, the purpose of which is to take advantage of the energy available in the reject water generated by the SWRO membranes. These systems, which can provide part of the energy that the SWRO system requires to desalinate the seawater, can be classified into two groups according to the basis on which they operate: centrifugal and isobaric energy–recovery systems.

- Membrane flushing and cleaning systems. The purpose of flushing systems is to automatically flush out concentrated water from the interior of the membranes to avoid their deterioration when the SWRO rack is in shutdown mode. For this, they use water previously generated by the SWRO system and stored in a tank specifically for this task. The aim of the cleaning system is to remove any impurities that may have accumulated on the membranes during their operation. This system basically consists of a tank in which the chemical cleaning reagents are mixed at the appropriate temperature, cartridge filters and a pump which discharges the correct flow to feed each pressure vessel. The cleaning system is normally used when membrane differential pressure rises by more than 10% over the original value or when productivity falls by more than 10% at constant temperature.
- Instrumentation and control system. The instrumentation of an SWRO system can vary from sophisticated to very simple systems. Sophisticated systems usually have automatic instrumentation with multiple sensors (for pressures, flow rates, temperatures, conductivities, feed water and product water turbidity levels, feed water and product water pH, levels of the different tanks, etc.), actuating devices (motorized valves, switches, variable frequency drives, etc.) installed in the different plant equipment (pump station, pre-treatment, SWRO systems, flushing and cleaning systems, post-treatment, product water storage, etc.), and control devices with programmable logic controllers (PLCs), industrial computers and supervisory control and data acquisition (SCADA) systems. Simpler systems have integrated instrumentation comprising two manual valves, two rotameters, a pressure manometer and a control which is limited to shut down tasks in the event of equipment failure to protect the main components of the system.

- Permeate treatment equipment, the purpose of which is to dose feed certain chemical products that enable conditioning (mineralization, neutralization, disinfection) of the permeate water prior to its distribution and end use.
- Product water pump station. The fresh water generated by the desalination plant is normally stored in a tank and subsequently pumped into other storage tanks or sent to the end users.
- System to discharge the reject water of the SWRO plant into the sea.
- Electrical energy supply system. Although SWRO plants have traditionally been powered by electrical energy generated from fossil fuels, the aim today is to reduce the economic and environmental costs caused by the high energy consumption of SWRO plants. Numerous proposals have been made for this purpose, and various types of renewable generation systems are installed in different parts of the world [20].

5.2.3 Some Observations on the Operating Parameters of an SWRO Plant

In this section, a description is given of the influence of various parameters on the operation of an SWRO plant, as well as an explanation of some of the restrictions imposed on these parameters. In this context, the theoretical models of the passage of permeate and salts through membranes [7, 12]. Equation 5.1 shows the productivity of an SWRO plant.

$$Q_p = K_w \cdot A \cdot \Delta P \tag{5.1}$$

where ΔP is the net pressure (which depends on the feed pressure, the osmotic pressure on the permeate side, the permeate pressure and pressure losses across the membranes), K_w is the water permeability coefficient and A is the surface area of the membranes. While both the feed and permeate pressure can be regulated by the plant operator within certain limits, the osmotic pressure depends on the quality of the feed water (molar concentrations of the diverse dissolved salts) and its temperature. An increase of the latter affects water permeability and the passage of salts across the membranes, increasing them without, generally, a beneficial end result.

Equation 5.2 shows the passage of salts across a membrane, Q_s.

$$Q_s = K_s \cdot A \cdot \Delta C \tag{5.2}$$

where ΔC is the mean difference between salt concentration on the two sides of the membrane, K_s is the salt transfer coefficient and A is the surface area of the membrane.

It should be noted that when the membranes reject the passage of salts, there is a resulting increase in salinity and impurities at the boundary layer near the membrane surface [7, 12]. This effect, known as concentration polarization, can result in the

concentration in that layer being far higher than the average concentration in the general brine flow. To avoid the harmful effects associated with this effect, the plant should not operate with polarization concentrations that entail increases above 13% with respect to the average brine concentration.

It can be deduced from Eqs. 5.1 and 5.2 that by varying the pressure and/or temperature the quantity and quality of the product water are modified for a given SWRO membrane configuration and a given feed water flow rate and characteristics.

For a specific feed water concentration and temperature, an SWRO membrane configuration and for a specific age of the elements and a given fouling factor, the parameters of feed water pressure (p_f) and flow rate (Q_f) can vary within an acceptable range. These parameters define the theoretical operating area of the SWRO plant, the limiting factors of which are maximum feed water flow rate, minimum brine flow of elements, maximum concentration of salts in the product water, minimum conversion rate (ratio between the product flow rate, Q_p, and the feed flow rate, Q_f), minimum brine to permeate ratio, and maximum average flux, Fig. 5.3 [22].

Each set of coordinates (Q_f, p_f) of this acceptable membrane system operating area defines a conversion rate, product water concentration, permeate flow (Q_p), power consumption and specific energy consumption of the desalination plant.
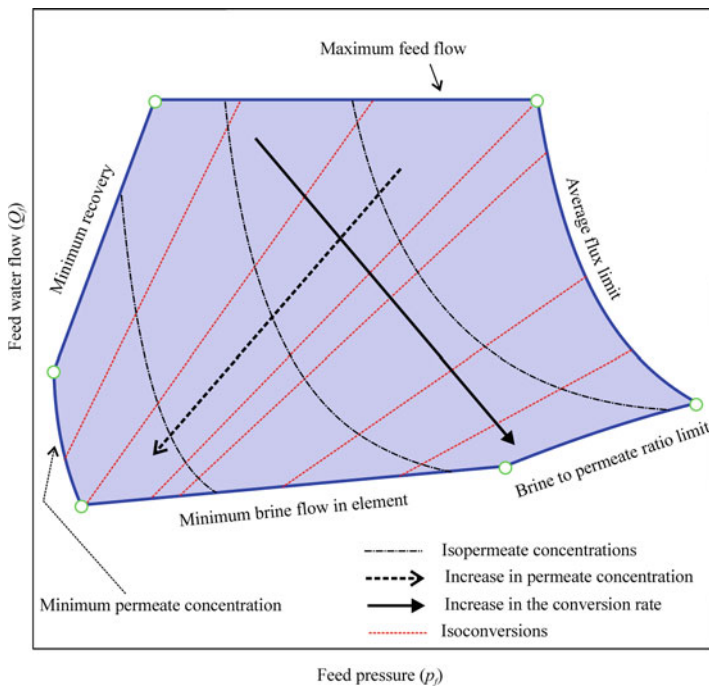
Fig. 5.3 Theoretical form of the operating area of an SWRO plant for a specific feed water conductivity and temperature, for a given SWRO membrane configuration and for a specific age of the elements and a given fouling factor

Given that the specific energy consumptions of an SWRO plant are directly proportional to the operating pressure and feed flow rate and inversely proportional to the product flow rate, the SWRO plant tends to operate with high conversion rates and use energy recovery systems with a view to reducing energy costs. However, high conversion rates result in an increase in brine salinity in the membranes with a consequent increase in osmotic pressure.

5.3 Overview of the Use of Computational Intelligence in the Desalination Industry

As previously mentioned, since 1993—when El-Hawary [13] published a proposal for possible application of AI techniques, and more specifically artificial neural networks (ANNs), in the desalination industry—several authors have carried out different studies in this field. In 1994, Rao et al. [14] carried out an in-depth study on the benefits of applying AI techniques in MSF desalination plant control systems and developed an ANN-based intelligent control hierarchy for such plants. These authors defined the different levels of an evolved intelligent automation system, detailed the problems at each of the control system levels, and explained the potential use of ANNs to resolve them. In their work, Rao et al. [14] proposed dividing the application of AI techniques in the desalination industry into four groups:

1. Monitoring and optimization,
2. Modelling and identification,
3. Adaptive control and
4. Conventional control systems.

To date, most of the relevant studies that have been published have used ANNs and have focused on the modelling, identification and simulation of desalination plants installed in various parts of the world (Table 5.1). In this context, Al-Shayji and Liu [15] used ANNs to model the operating performance of two large-scale commercial desalination plants; an MSF desalination plant located in Kuwait with a capacity of 181,760 m^3/day and an SWRO plant in Saudi Arabia with a capacity of 56,800 m^3/day. Along the same lines, Jafar and Zilouchian [23] estimated the product flow rate and the amount of total dissolved solids (TDS) of two RO desalination plants which used brackish water and seawater, respectively. Murthy and Vora [24] estimated the reject and product water flow rate of an RO desalination plant. Abbas and Al-Basaki [25] used feed water temperature, pressure and TDS values to obtain the product flow rate of an RO unit. Lee et al. [26] also used the feed water, TDS and flow rate values, but additionally incorporated data on the pressure gradient across the membrane (trans-membrane pressure—TMP). Using these data, they estimated the TDS and product water flow rate of an SWRO plant located in the United Arab Emirates and designed to produce 464,000 m^3/day. Similarly, Libotean et al. [27] also estimated these two variables (TDS and product flow rate) but did so employing two timescales, one for the current period and one for 24 h forecast horizons.

Table 5.1 Studies which have used computational intelligence in the desalination industry

Topic of the study	Year	Ref.
Artificial neural networks and possible applications to desalination	1993	[13]
Towards improved automation for desalination processes. Intelligent control	1994	[14]
Neural networks for the identification of MSF desalination plants	1995	[16]
Prediction of critical desalination parameters using RBF networks	2002	[23]
Predictive modelling of large-scale commercial water desalination plants	2002	[15]
Prediction of reverse osmosis performance using ANN	2004	[24]
Modelling of an RO water desalination unit using ANN	2005	[25]
ANN model for optimizing the operation of an SWRO desalination plant	2009	[26]
ANN approach for modelling the performance of reverse osmosis membrane desalting	2009	[27]
Prediction of temperature elevation for seawater in MSF desalination plants using RBF neural networks	2010	[17]
Prediction of the dialysis process performance using ANNs	2011	[18]
ANN-based correlation for estimating water permeability constant in RO desalination process under fouling	2014	[30]
Modelling, optimization and control of RO water treatment in Kazeroon power plant using ANN	2015	[28]
ANN approach for predicting RO desalination plants performance in the Gaza Strip	2015	[29]
Modelling and simulation of VMD desalination process by ANN	2016	[19]
ANN applied to manage the variable operation of a simple seawater reverse osmosis plant	2017	[36]
Modelling fouling in a large RO system with ANN	2018	[31]
Wind-driven SWRO desalination prototype with and without batteries	2018	[37]

For this, they used the variables of feed flow rate, conductivity, pressure, pH and temperature. Barrello et al. [28] suggested the use of ANNs for a different purpose, namely estimating the water permeability constant in an RO desalination system for any salinity and operating pressure under fouling. Madaeni et al. [29] modelled the process of three RO plants using ANNs for the long-term forecasting of performance degradation (TMP, TDS and product flow rate) and used a genetic algorithm (GA) to find the optimum paths of TMP, feed flow rate and control strategies during a specific period of time. Aish et al. [30] implemented two multilayer perceptron (MLP) architectures to predict the performance of five RO desalination plants on the Gaza Strip which used brackish water. With one MLP architecture, the authors estimated final TDS from the feed water temperature, pressure, pH and TDS values. With the other, they estimated the product flow rate using as input data the feed water pressure, pH and TDS values. Roehl et al. [31] proposed the use of ANNs for membrane fouling modelling in a large RO system of 284,000 m^3/day. These authors predicted early fouling and later fouling with data taken over 6 years of numerous variables
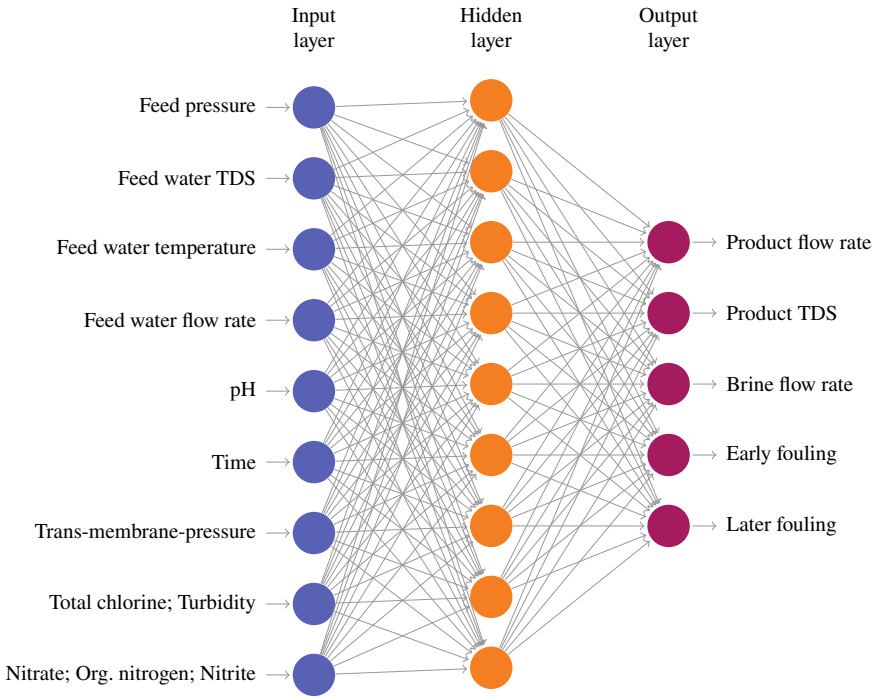
Fig. 5.4 Commonly used input and output variables in studies which apply ANNs in desalination plant modelling

including total chlorine, electrical conductance, TDS, turbidity and nitrate, organic nitrogen and nitrite content [31].

All the authors mentioned in this section concur that ANNs offer sufficiently satisfactory results for their application in the modelling, online control, forecasting and simulation of an RO desalination system [15, 23–31]. If all the input and output variables used by the different authors were to be combined in a single ANN, the configuration shown in Fig. 5.4 would be the result. The variables are represented in Fig. 5.4 by order of use (the most used variable is shown at the top and the least used at the bottom). As can be seen, the vast majority of authors use an ANN with a single hidden layer which uses, as input variables, the data values of feed water pressure, TDS, temperature and flow rate and, as output variables, product water flow rate and TDS.

5.4 Computational Intelligence in Desalination with Renewable Energies

In order to show the potential uses of CI techniques in desalination with renewable energies and the degree of difficulty that the challenge of their implementation can entail, the following subsections will first offer a succinct description of the most common types of renewable energy-powered desalination systems that have been proposed or implemented in the world to date [20]. Although the number of proposed renewable energy sources is extensive, this chapter, without loss of generality, will focus on wind energy as it is one of the most commonly used sources because of its technological maturity and the competitive cost of the electrical energy produced with it. Following this, a description will be given of the wide variety of potential uses of CI techniques in the different subsystems and tasks that are carried out in the operation of wind energy-powered desalination systems. Detailed descriptions of CI techniques and methodologies are not offered here but can be consulted in the extensive specialist literature [32–35].

Finally, a summary will be given of the results obtained to date, as well as further studies pending development, with a small-scale prototype SWRO desalination plant. This plant, in which CI techniques have been implemented [22, 36, 37] in the control system, has been designed for continuous adjustment of its energy consumption to the widely varying power generated by a stand-alone wind turbine. It is currently being tested on the island of Gran Canaria (Spain).

5.4.1 Classification of Wind Energy-Powered Desalination Systems

Wind energy-powered desalination systems are generally classified into two major groups [20], which can in turn be subdivided into various subgroups, Fig. 5.5.

1. Systems in which the desalination and renewable electrical energy generation technologies are connected to conventional grids. For purposes of differentiation, these can be termed 'on-grid systems'. Within this group, and depending on the manner in which the electrical energy that is generated by the renewable technology is managed, two different types can be distinguished: (a) systems in which all the renewable-sourced electrical energy is fed into the conventional grid, with the desalination plants powered, like any other load of the system, by the electrical energy transported by the grid; (b) systems in which the renewable-sourced electrical energy is consumed in the first place by the desalination plants and the energy surplus (or deficit) is fed into (or extracted from) the conventional grid.
2. Stand-alone and hybrid systems which supply the electrical energy required by the desalination plants that are connected to them. For purposes of differentiation,
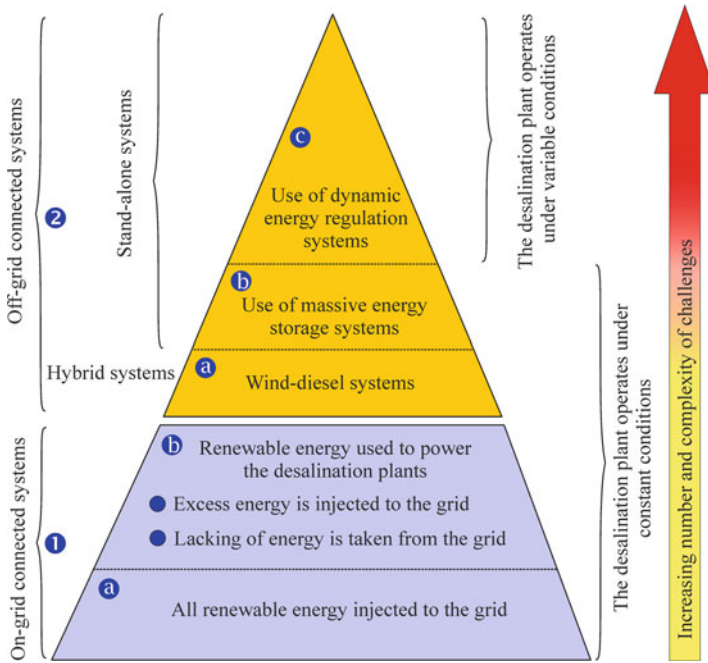
Fig. 5.5 Classification of wind-powered desalination systems and levels of challenge in the implementation of computational intelligence

these can be termed 'off-grid systems'. Three types can be distinguished within this group: (a) systems in which the desalination plants operate under rated conditions and in which, when problems arise as a result of the variable nature of the renewable energy source, hybrid energy solutions (e.g. a wind–diesel system) are employed; (b) systems in which the desalination plants also operate under rated conditions but which use massive energy storage devices (generally batteries) to counter the effects caused by the variable nature and uncertainty of the renewable energy source and to ensure adaptation of the fresh water supply to demand; (c) systems in which the desalination plants operate under variable conditions, with dynamic modification of the number of SWRO systems that need to be operating [38], as well as their operating parameters [22, 36, 39], in order to continuously adapt the energy consumption of the desalination plant to the renewable-sourced electrical energy that is available at each instant. Note that these latter systems usually have dynamic power regulation systems (e.g. ultracapacitors bank, flywheel, etc.) to alleviate the problems of instability that can appear.

The above groups and subgroups are represented in Fig. 5.5 in a hierarchical pyramid; the systems with the highest installed worldwide capacity to date are shown at the base of the pyramid and those with the lowest installed capacity at its apex.

5.4.2 Potential Challenges in the Implementation of CI Technique in Wind Energy-Powered Desalination Systems

All the above groups and subgroups of wind energy-powered desalination systems can employ CI techniques. However, the number of challenges involved in the implementation of such techniques in these systems, as well as their degree of complexity, differs depending on the level of uncertainty that is introduced in the operation by the random nature of the renewable energy. The number of challenges will thus effectively depend on the wind energy-powered desalination system subgroup type. In this context, and by way of reference, the pyramid of Fig. 5.5 hierarchically shows the increase and level of challenge complexity. These challenges need to be tackled for the implementation of CI techniques in each wind energy-powered desalination system group and subgroup. The systems with relatively minor challenges and difficulties for the implementation of CI techniques are shown at the base of the pyramid. As the pyramid rises, the challenges become more numerous and more complex. As can be seen in Fig. 5.5, on-grid connected systems present relatively minor CI implementation challenges. This is essential because such systems, in which the desalination plants operate under rated conditions, can normally rely on relatively strong grids which are usually only very slightly affected by the uncertainty generated by the renewable energy source. If the grids of such systems are weak, the penetration of renewable energies is generally limited to avoid problems of instability. It can also be seen in Fig. 5.5 that the use of CI techniques implies more numerous and more complex challenges when managing and controlling the different subsystems that are integrated into the group of stand-alone and hybrid systems. This is especially true for subgroup 2c, due to the complexity entailed by desalination plant operation under variable conditions and the corresponding high level of uncertainty.

A schematic representation is shown in Fig. 5.6 of the general configuration of a stand-alone wind-driven SWRO desalination system (valid to represent both subgroup 2b and 2c). Shown in Fig. 5.6 is a list of potential applications of CI techniques (provided by the AI system) in the different subsystems that make up the microgrid: renewable generation subsystem (wind energy conversion system, in this case), desalination subsystem (Fig. 5.2), energy storage and dynamic power regulation subsystem, and control subsystem.

Renewable generation subsystem

Wind turbines can be classified by their rated power as small, medium and large [20]. While the trend in on-grid systems is to use wind turbines with ever higher rated powers, small wind turbines have been the most commonly used in stand-alone systems [20], though medium wind turbines have also been employed [38, 39]. The technological innovations (variable-speed wind turbine rotors, full span pitch control, active stall control, active yaw control, power converters, etc. [20]) which have been implemented in medium and large-sized turbines, and which enable their increasingly optimal operation, have not been incorporated in the small-sized ver-
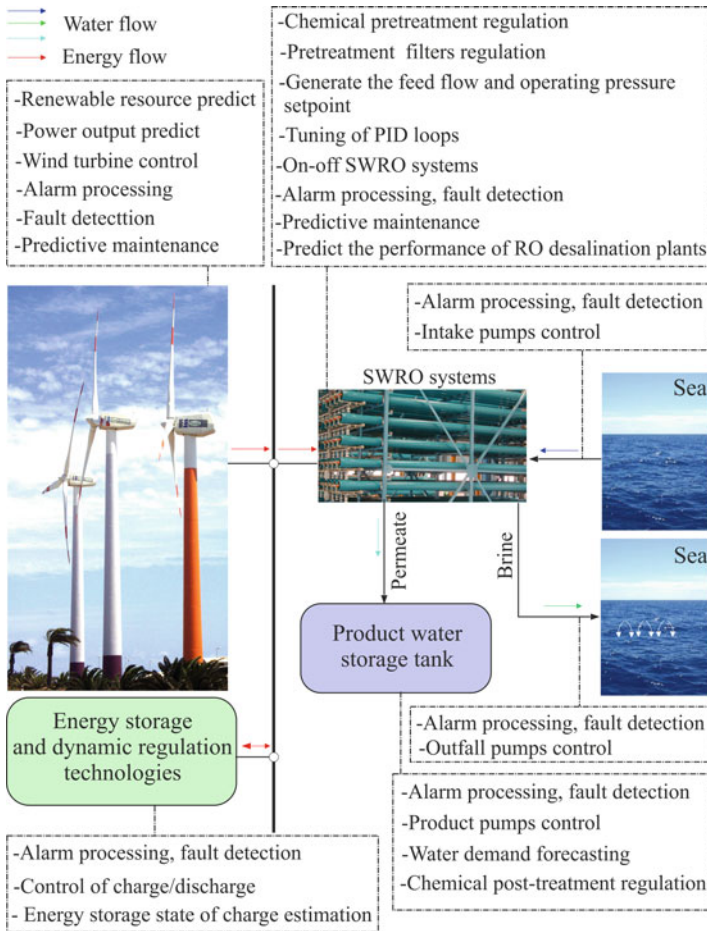
Fig. 5.6 Potential applications of CI techniques in the different stages of renewable energy-powered desalination systems

sions [40]. Small wind turbines tend to have regulation and control systems (passive stall control, passive pitch control, passive yaw control, etc. [20]) that are extremely limited. CI techniques can be used in all types of wind turbines for alarm processing, fault detection or predictive maintenance tasks [37–44]. However, the extent of their usefulness in regulation and control tasks depends on the degree of complexity of the regulation and control systems that are incorporated in the wind turbines. CI techniques can be used to participate in the regulation of power converters and in the optimization of control system parameters, including those of the yaw and pitch control systems [44–47]. Yaw is the angle that the nacelle of the wind turbine is made to rotate through with respect to the vertical axis of the tower that supports it. Controlling this angle is essential to ensure that the rotor of the wind turbine is

always facing the direction of the wind. Pitch is the angle that the blades of a wind turbine are made to rotate through with respect to their longitudinal axes to control the power of the turbine. Controlling this angle is essential to maintain an optimum pitch angle that ensures maximum efficiency of the wind turbine at all wind speeds.

Independently of the type of wind turbine that is employed, CI techniques can be used in forecasting and prediction tasks with respect to the renewable resource and the power produced by the wind turbines [44–50]. Such forecasts can be made in time horizons that can be very short term (a few seconds to up to 30 min), short term (30 min to 6 h), medium term (6–24 h) and long term (1–7 days) [44, 48, 49]. With them, it is possible to anticipate the orders of the global control system, appropriately respond to generation variations, adequately plan maintenance tasks, and establish optimum energy management strategies [37].

Desalination subsystem

CI techniques can be used in the desalination subsystem independently of the group or subgroup in which it is incorporated, Fig. 5.5, and in each and every one of its components, Fig. 5.2, in alarm processing, fault detection and predictive maintenance tasks, Fig. 5.6. In this respect, these techniques can be extremely effective in the early detection of possible membrane scaling, fouling and deterioration, and to give notification of the ideal moment for membrane cleaning, etc. The bullet points shown below give an overview of the potential applications of CI techniques in the different operating stages of an SWRO desalination plant:

- control of the VFDs of the seawater intake pumps; the aim is to ensure an optimum flow of pumped water, especially when the SWRO desalination plant is operating under variable conditions (modifying the number of SWRO systems in operation and/or the parameters of the feedwater flow and/or their operating pressure).
- optimal regulation of the physico-chemical seawater pre-treatments. This is generally performed as and when required on the basis of the experience of the workers. However, given the optimizing and decision-making capacity of some CI techniques—as with GA or multivariable regression techniques—they can be used to carry out such functions which depend on multiple operating parameters of an SWRO desalination plant.
- optimal control of the desalination process, especially in stand-alone systems which operate by varying the number of connected/disconnected SWRO systems [38] and/or the operating pressure and/or feed flow parameters [22, 36, 39] in order to adapt the energy consumption of the SWRO desalination plant to the renewable energy sourced electrical energy that is available at each instant. In this case, CI techniques can be used to determine the number of SWRO systems that need to be operating at each moment and to generate the feed flow and operating pressure setpoints that the control system should use at each instant depending on the electrical energy that is available and the characteristics of the feedwater (conductivity, temperature, etc.) [36]. The CI techniques that are used have to take into account the operating restrictions of the system, Fig. 5.3, and generate the control setpoints following a certain strategy: constant conversion rate (variable operating pressure

and feed flow), constant product water quality (variable operating pressure and feed flow), constant operating pressure and variable feed flow, variable operating pressure and constant feed flow, etc. [51]. These techniques are also suitable candidates for the implementation of adaptive proportional–integral–derivative (PID) controllers. These can be used to improve the optimal adjustment of parameters to the complexity of a system which is not linear and whose parameters may need to be modified due to abrupt changes that can arise in the amount of energy that is available and that can be accompanied by abrupt changes in operating pressure and/or feed flow [52–55].

- forecasting of the operation of the SWRO desalination plant. CI techniques can be used especially in predicting the obtainable product water flow and its quality. It should be noted that these tasks are the most commonly performed ones to date [23–26, 37], as mentioned in Section 5.3 of this chapter.
- control of the VFDs of the pumps that discharge into the sea the reject water of the SWRO plant; their purpose is to regulate this flow, especially when the SWRO desalination plant is operating under variable conditions resulting in reject flows which vary over time.
- optimal regulation of the dosing pumps of the chemical products that are required to prepare the water produced by the SWRO desalination plant.
- control of the VFDs of the supply pumps of the product water stored in a tank to cover demand. CI techniques can also be used to predict water demand in different time horizons [56–58] and, based on such predictions and renewable energy availability predictions, to optimize global system operation.

Energy storage and dynamic power regulation subsystems

As with the other subsystems, CI techniques can be used in tasks related to alarm processing, fault detection and predictive maintenance of the energy storage devices and in power regulation devices, Fig. 5.6. Energy storage devices are generally comprised of batteries and used in subgroup 2b of stand-alone wind-driven desalination systems, Fig. 5.5. Their main purpose is to adapt the wind energy supply to the energy demand of the SWRO desalination plant operating under constant conditions to ensure a given freshwater demand can be met. Power regulation devices are generally comprised of flywheels and supercapacitors given their rapid response to variations in power. They are proposed for use in subgroup 2c of stand-alone wind-driven desalination systems, Fig. 5.5, for the purpose of smoothing out the instantaneous differences that can arise between the power generated by the wind turbines and the power demanded by the SWRO desalination plant operating under variable conditions.

In addition, these techniques can be used in battery and supercapacitor charge/discharge control tasks, in battery state-of-charge estimation [59–63], in flywheel charge/discharge control [64], etc.

Control subsystem

CI techniques can be used in the global control system of stand-alone wind-driven desalination systems to manage and control their operation according to the results

obtained by the CI techniques applied in each subsystem and each operating stage. The challenges associated with the management and global control of the system are considerable. This is especially true when the intelligent management and control are being pursued of a complex load—powered by a renewable energy source that varies over time and displays uncertainty—in a weak grid. Unlike stronger electrical power systems, a weak grid experiences more changes in voltage and frequency parameters that need to be dealt with. It should be noted that the degree of difficulty in terms of overcoming the challenges faced is related to, among other factors, the technological level of the renewable generation system. In older technological generation systems control and management are centred on the load (desalination plant), whereas in more advanced generation systems control and management are distributed between generation and load. Though system management and control become easier as an increasing number of technological advances are incorporated in the generation system, there is a corresponding increase in the number of parameters that the CI techniques have to manage.

5.4.3 Results Obtained with a Small-Scale Prototype SWRO Desalination Plant Controlled Using CI Techniques

On the island of Gran Canaria (Spain), a small-scale prototype SWRO desalination plant has been put into operation, designed for continuous adjustment of its energy consumption to the widely varying power generated by a stand-alone wind turbine. CI techniques have been incorporated in the control system of this prototype [36, 37], which is currently being tested, Fig. 5.7.

The system, whose components are described in detail in reference [22], belongs to subgroup 2c of the stand-alone systems, Fig. 5.5. One of the components is a 15 kW wind turbine with limited regulation and control capacities as it only has passive pitch and passive yaw control (specifically a wind vane). Therefore, the management and control that are required to balance the energy generated by the wind turbine and the energy demanded by the SWRO desalination plant fundamentally focus on the latter. The function of the ultracapacitors bank is to dynamically balance the power differences that arise between the electrical power generated by the wind turbine and the power demand of the SWRO desalination plant.

A simplified outline is given in Fig. 5.8 of the prototype SWRO desalination plant. Only the components that are required for an understanding of the operation of CI models implemented in the plant are shown.

The desalination plant, which has no energy recovery system, has an SWRO system which is comprised of a high-pressure feed pump (reciprocating pump) with a VFD that allows it to work with a feed flow range of between 1 and 6 m^3/h, and a single-pass SWRO system with membrane flushing and cleaning systems. The rack of the SWRO system supports two pressure vessels, each of which has three membranes (spiral-wound polyamide type) connected in series.
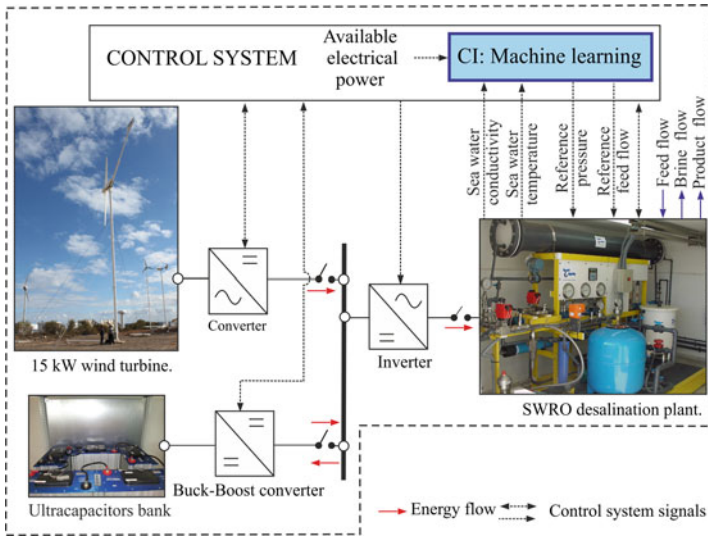
Fig. 5.7 Prototype SWRO desalination plant. *Source* Figure adapted from [36]

The SWRO desalination plant has multiple sensors (operating pressure, feed and product flow rates, feedwater temperature, feed and product water conductivities), actuating devices and a control system with PLCs, a personal computer and an SCADA system.

Among the actuating devices, the SWRO system has a set of valves, Fig. 5.8, which allow the control system to connect one or both pressure vessels (hydraulically in parallel) to roughly adapt the power demand to the wind power supply depending on the range of availability of the latter. To obtain a narrower adjustment in the adaptation of the power demand of the SWRO desalination plant to the wind power supply, the control system uses, on the one hand, the VFD of the high-pressure pump to modify the feed flow and, on the other hand, a valve which regulates the operating pressure of the membrane system, Fig. 5.8.

The flows and operating pressures are restricted to an operating area [36] similar to the one represented in Fig. 5.3 for a specific feed water concentration and temperature, an SWRO membrane configuration, a specific age of the elements, and a given fouling factor. Of the variable operating strategies of the SWRO desalination plant that have been proposed [51], it was decided to implement the one in which the control system varies the operating parameters in such a way that the conversion rate remains constant, specifically at an approximate value of 13.5%. This conversion rate is obtained at the intercept of the average flux limit line and the maximum feed seawater flow line [37], Fig. 5.3. With this conversion rate, it is possible to obtain the highest range of variation of power of the wind turbine that can be used by the SWRO desalination plant to enable it to operate without interruption.
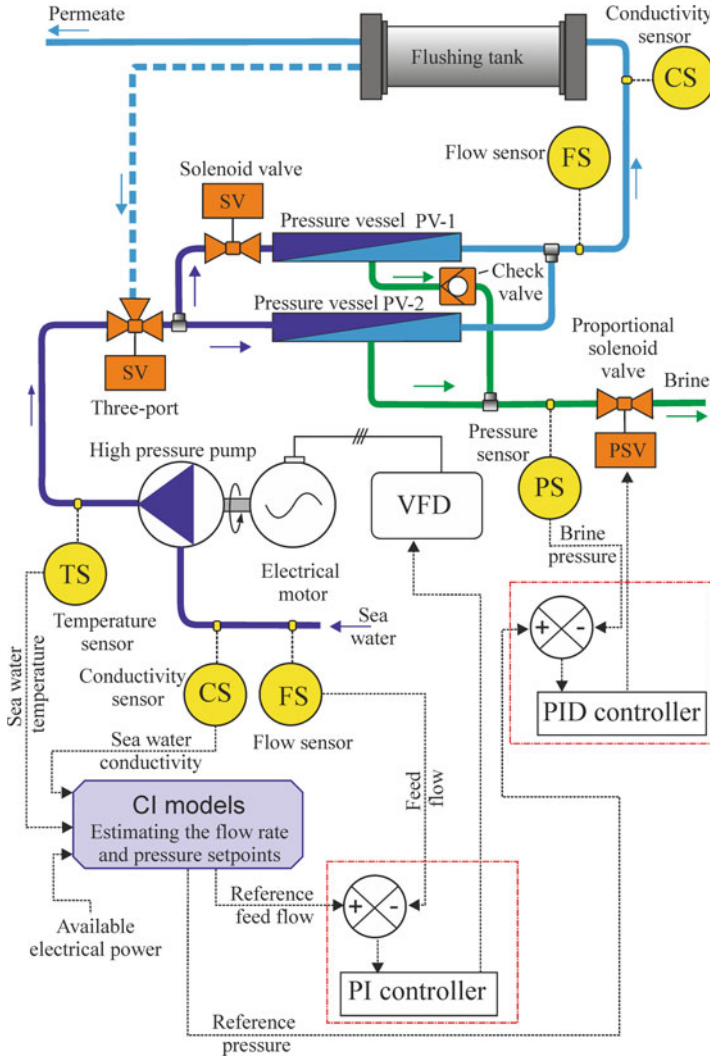
Fig. 5.8 Simplified outline of the prototype SWRO desalination plant and its control system

CI techniques have been incorporated in the control system to enable it to manage the feed flow and operating pressure setpoints of two closed-loop feedback controls which regulate the actuating devices of the SWRO desalination plant corresponding to these two variables (closed-loop PID for the case of pressure and closed-loop proportional–integral for the case of flow, Fig. 5.8). These CI techniques are, more specifically, ANN algorithms [36, 37], support vector machine (SVM) [37] and random forest (RF) [37].

For the design of the models based on ANN architectures, which have been the most commonly used type to date, a new method was used based on a real-coded genetic algorithm (called MI-LXPM) and the resilient backpropagation (RPROP) algorithm [36]. The proposed method establishes a criterion to determine the number of hidden layers of neurons and of the number of neurons of each hidden layer in the ANN models. In other words, one of the aims of the method is to determine whether more than one hidden layer of neurons is unnecessary in most of the problems, as has been theoretically demonstrated [65, 66], or whether in fact two layers of hidden neurons are required, as has been proposed by various authors [23–25, 29, 30]. A simple block diagram is shown in Fig. 5.9 of the design method used.

Operating data, using both one and two pressure vessels, of the SWRO plant have been collected and used to train the CI models (ANN, SVM, RF). The input parameters of the CI models have to date been the available wind powers and the feedwater conductivities and temperatures for a specific age of the elements and a given fouling factor. In other words, the influence of these latter two variables has not been considered.

As well as machine learning models trained and implemented in the control system to manage the feed flow and operating pressure setpoints of the SWRO desalination plant, models have also been developed to predict product water flow and conductivity [37].

According to the authors of the tests that have been performed thus far [36, 37], the analysis undertaken has demonstrated the ability of these machine learning models to manage the operating setpoints of the SWRO desalination plant and to successfully adapt the energy consumption of the plant to the wide and random variation of the available electrical power.

Shown in Fig. 5.10 is the behaviour of the powers consumed by the SWRO plant (controlled by a CI model based on ANNs which generate the operating setpoints) when operating with two differing seawater characteristics (conductivity and temperature). The statistical test that was performed shows no significant statistical differences (at 5% level) between the MAE (mean absolute error) and MAPE (mean absolute percentage error) committed when adapting power consumption of the plant to the available electrical power in the various tests performed using differing feedwater characteristics.

In addition, the authors [37] evaluated and compared the performance of the three previously mentioned machine learning techniques (ANN, SVM and RF) in predicting the variables operating pressure (p_f), feed flow (Q_f), permeate flow (Q_p) and permeate conductivity (C_p). They used a statistical procedure based on cross-validation to obtain the values of the MAE and MAPE metrics for each of the estimated variables and applied a statistical significance test to compare the results of the three techniques. It was concluded from an analysis of the results that the MAE and MAPE errors obtained with the ANN techniques were significantly higher (5% significance level) than those produced with the SVMs and RFs. After applying a tenfold cross-validation, it was also deduced that the RF technique obtained better mean values of the metrics than the SVM technique [37]. For these reasons, the

START

Generation of initial population.

(Numbers of neurons in each hidden layer)

Create cross validation partition data
Total number of datasets randomly ordered

	Fold-1	Fold-2	Fold-3	Fold-4	Fold-5	Fold-6	Fold-7	Fold-8	Fold-9	Fold-10

Data set-1
Data set-2
Data set-3
Data set-i
Data set-8
Data set-9
Data set-10

Training and validation sets   Test set

Loop: Do i=1,...,10

Function fitting neural network   Test set

Data set-i

Training and validation sets

Resilient backpropagation algorithm (**trainrp**)

Update of weight
and bias values

Training set (80%)   Training → Validation   Training stop   Model i

Learning model

Validation set (10%)

Testing model-i

Errors i: MAE_i

$$\overline{MAE} = \frac{1}{10} \sum_{i=1}^{10} MAE_i$$

Is stopping criteria satisfied?   No   Creating 40 new subsets of neurons in each hidden layer by applying Selection, Crossover and Mutation to current population

Yes

Knowledge of:
- Numbers of hidden layers.
- Numbers of neurons in each hidden layer.
- Errors: $\overline{MAE}$.

END

Fig. 5.9 Simplified block diagram of the method used for the design of the ANNs
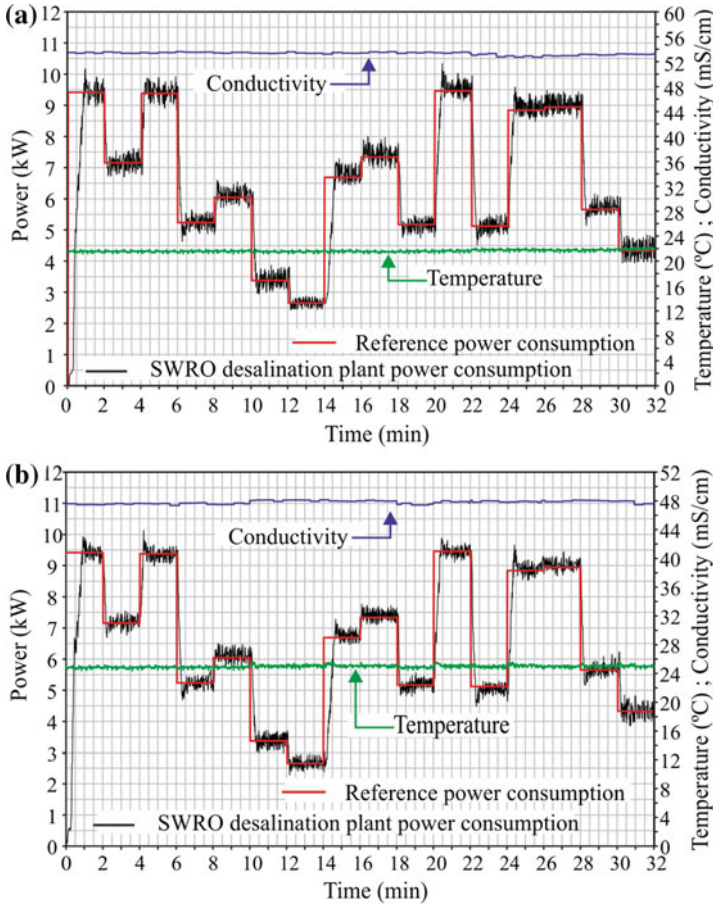
Fig. 5.10 Examples of results of powers consumed by the SWRO plant (in black) obtained when regulating the feed flow and operating pressure setpoints through ANNs to match the variable reference powers (in red) with two different levels of seawater characteristics: **a** results for seawater conductivities between 52.7 and 53.6 mS/cm and seawater temperatures between 21.1 and 22.3 °C; **b** results for seawater conductivities between 50.5 and 51.0 mS/cm and seawater temperatures between 22.8 and 23.9 °C

authors recommend the use of the SVM and RF techniques (especially the latter) to predict SWRO desalination plant performance rather than the ANN technique [37].

Although the potential uses of CI techniques in wind-powered desalination systems are wide-ranging, to date very few studies have been published on the implementation of these techniques in such systems Nonetheless, the results obtained with the prototype described above are promising and the prototype itself has been shown to be a good candidate to test the benefits of CI techniques. As a result, the analysis of the implementation of most of the potential applications indicated in Section 5.4.2 of this chapter is currently underway. More specifically, implementa-

tion of CI techniques has been initiated to replace the PID controllers with controllers which use multiple input variables, and which are capable of anticipating phenomena detectable by these techniques [37]. Models are also being developed for the forecasting of available energy and for the management and modelling of dynamic energy storage systems.

5.5 Conclusions

This chapter has offered an overview of the potential applications of CI techniques in the desalination industry, with particular reference to wind energy-powered SWRO desalination plants. An indication has also been given of the degree of difficulty of the challenges entailed in the implementation of CI techniques in the most commonly used types of renewable energy-powered desalination systems. A brief description has also been given of the results obtained to date, as well as the works pending development, from a small-scale prototype SWRO desalination plant designed for continuous adjustment of its energy consumption to the widely varying power generated by a stand-alone wind turbine. CI techniques have been incorporated into the control system of this prototype and tests are presently underway on the island of Gran Canaria (Spain). It has been shown that CI can be efficiently used in systems with complex dynamics, as is the case of SWRO desalination plants. Such plants are not linear and, when directly powered by renewable energies, vary over time with uncertainty and, therefore, are extremely difficult to control with conventional methods.

References

1. Clarke, R.: Water: The International Crisis. Routledge (2013)
2. Gleick, P.: World's Water 200–2001. Island Press (2000)
3. Petrella, R.: The Water Manifesto: Arguments for a World Water Contract. Zed Books (2001)
4. Falkenmark, M.: Population, environment and development: a water perspective. In: United Nations Expert Group Meeting on Population, Environment and Development United Nations, New York, USA, pp. 99–116 (1994)
5. Seckler, D., Barker, R., Amarasinghe, U.: Water scarcity in the twenty-first century. Int. J. Water Resour. Dev. **15**, 29–42 (1999)
6. Mandil, M.A., Bushnak, A.A.: Future needs for desalination in South Mediterranean countries. Desalination **152**, 15–18 (2003)
7. Kucera, J.: Desalination: Water from Water. Wiley (2014)
8. Subramani, A., Jacangelo, J.G.: Emerging desalination technologies for water treatment: a critical review. Water Res. **75**, 164–187 (2015)
9. Youssef, P.G., Al-Dadah, R.K., Mahmoud, S.M.: Comparative analysis of desalination technologies. Energy Procedia **61**, 2604–2607 (2014)
10. Khawaji, A.D., Kutubkhanah, I.K., Wie, J.: Advances in seawater desalination technologies. Desalination **221**, 47–69 (2008)

11. Delyannis, E., Belessiotis, V.: Desalination: the recent development path. Desalination **264**, 206–213 (2010)
12. Voutchkov, N.: Desalination Engineering: Planning and Design. McGraw-Hill (2013)
13. El-Hawary, M.E.: Artificial neural networks and possible applications to desalination. Desalination **92**, 125–147 (1993)
14. Rao, G.P., Al-Gobaisi, D.M.K., Hassan, A., Kurdali, A., Borsani, R., Aziz, M.: Towards improved automation for desalination processes. Part II: intelligent control. Desalination **8**(97), 507–28 (1994)
15. Al-Shayji, K.A., Liu, Y.A.: Predictive modeling of large-scale commercial water desalination plants: data-based neural network and model-based process simulation. Ind. Eng. Chem. Res. **41**, 6460–6474 (2002)
16. Selvaraj, R., Deshpande, P.B., Tambe, S.S., Kulkami, B.D.: Neural networks for the identification of MSF desalination plants. Desalination **101**, 185–193 (1995)
17. Aminian, A.: Prediction of temperature elevation for seawater in multi-stage flash desalination plants using radial basis function neural networks. Chem. Eng. J. **162**, 552–556 (2010)
18. Godini, H.R., Ghadrdan, M., Omidkhah, M.R., Madaeni, S.S.: Part II: prediction of the dialysis process performance using artificial neural network (ANN). Desalination **265**, 11–21 (2011)
19. Cao, W., Liu, Q., Wang, Y., Mujtaba, I.M.: Modeling and simulation of VMD desalination process by ANN. Comput. Chem. Eng. **84**, 96–103 (2016)
20. González, J., Cabrera, P., Carta, J.A.: Wind energy powered desalination systems. In: Desalination: Water from Water, pp. 567–646. Wiley (2018)
21. Voutchkov, N.: Desalination Engineering: Operation and Maintenance. McGraw-Hill (2014)
22. Carta, J.A., González, J.A., Cabrera, P., Subiela, V.J.: Preliminary experimental analysis of a small-scale prototype SWRO desalination plant, designed for continuous adjustment of its energy consumption to the widely varying power generated by a stand-alone wind turbine. Appl. Energy **137**, 222–239 (2015)
23. Jafar, M.M., Zilouchian, A.: Prediction of critical desalination parameters using radial basis functions networks. J. Intell. Robot. Syst. **34**(2), 219–230 (2002)
24. Murthy, Z.V.P., Vora, M.M.: Prediction of reverse osmosis performance using artificial neural network. Indian J. Chem. Tech. (IJCT) **11**(1), 108–115 (2004)
25. Abbas, A., Al-Bastaki, N.: Modeling of an RO water desalination unit using neural networks. Chem. Eng. J. **114**(1–3), 139–143 (2005)
26. Lee, Y.G., Lee, Y.S., Jeon, J.J., Lee, S., Yang, D.R., Kim, I.S., Kim, J.H.: Artificial neural network model for optimizing operation of a seawater reverse osmosis desalination plant. Desalination **247**(1–3), 180–189 (2009)
27. Libotean, D., Giralt, J., Giralt, F., Rallo, R., Wolfe, T., Cohen, Y.: Neural network approach for modeling the performance of reverse osmosis membrane desalting. J. Membr. Sci. **326**, 408–419 (2009)
28. Madaeni, S.S., Shiri, M., Kurdian, A.R.: Modeling, optimization, and control of reverse osmosis water treatment in Kazeroon power plant using neural network. Chem. Eng. Commun. **202**, 6–14 (2015)
29. Aish, A.M., Zaqoot, H.A., Abdeljawad, S.M.: Artificial neural network approach for predicting reverse osmosis desalination plants performance in the Gaza Strip. Desalination **367**, 240–247 (2015)
30. Barello, M., Manca, D., Patel, R., Mujtaba, I.M.: Neural network based correlation for estimating water permeability constant in RO desalination process under fouling. Desalination **345**, 101–111 (2014)
31. Roehl, E.A., Ladner, D.A., Daamen, R.C., Cook, J.B., Safarik, J., Phipps, D.W., Xie, P.: Modeling fouling in a large RO system with artificial neural networks. J. Membr. Sci. **552**, 95–106 (2018)
32. Sharma, K.D., Chatterjee, A., Rakshit, A.: Intelligent Control: A Stochastic Optimization Based Adaptive Fuzzy Approach. Springer-Verlag GmbH (2018)
33. Siddique, N.: Intelligent Control: A Hybrid Approach Based on Fuzzy Logic, Neural Networks and Genetic Algorithms. Springer (2016)

34. Nakamatsu, K., Kountchev, R.: New Approaches in Intelligent Control: Techniques, Methodologies and Applications. Springer (2018)
35. Liu, J.: Intelligent Control Design and MATLAB Simulation. Springer (2017)
36. Cabrera, P., Carta, J.A., González, J., Melián, G.: Artificial neural networks applied to manage the variable operation of a simple seawater reverse osmosis plant **416**, 140–156 (2017)
37. Cabrera, P., Carta, J.A., González, J., Melián, G.: Wind-driven SWRO desalination prototype with and without batteries: a performance simulation using machine learning models. Desalination **435**, 77–96 (2018)
38. Carta, J.A., González, J., Subiela, V.J.: Operational analysis of an innovative wind powered reverse osmosis system installed in the Canary Islands. Sol. Energy **75**, 153–168 (2003)
39. Paulsen, K., Hensel, F.: Introduction of a new energy recovery system-optimized for the combination with renewable energy. Desalination **184**, 211–215 (2005)
40. Association, European Wind Energy: Wind Energy—The Facts. Earthscan, London (2009)
41. Bangalore, P., Patriksson, M.: Analysis of SCADA data for early fault detection, with application to the maintenance management of wind turbines. Renew. Energy **115**, 521–532 (2018)
42. Pires, G., Mauricio, A., Carvalho, P.: Prognostic techniques applied to maintenance of wind turbines: a concise and specific review. Renew. Sustain. Energy Rev. **81**, 1917–1925 (2018)
43. Zhao, H., Liu, H., Hu, W., Yan, X.: Anomaly detection and fault analysis of wind turbine components based on deep learning network. Renew. Energy **127**, 825–834 (2018)
44. Pliego, A., García, F.P., Pinar, J.M., Ruiz-Hernández, D.: A survey of artificial neural network in wind energy systems. Appl. Energy **228**, 1822–1836 (2018)
45. Bilal, A., Liu, X.: Adaptive neuro-fuzzy algorithm to estimate effective wind speed and optimal rotor speed for variable-speed wind turbine. Neurocomputing **272**, 495–504 (2018)
46. Song, D., Fan, X., Yang, J., Liu, A., Chen, S., Hoon, Y.: Power extraction efficiency optimization of horizontal-axis wind turbines through optimizing control parameters of yaw control systems using an intelligent method. Appl. Energy **224**, 267–279 (2018)
47. Morshedizadeh, M., Kordestani, M., Carriveau, R., Ting, D.S.K., Saif, M.: Application of imputation techniques and adaptive neuro-fuzzy inference system to predict wind turbine power production. Energy **138**, 394–404 (2017)
48. Soman, S.S., Zareipour, H., Malik, O.: Mandal: P: A review of wind power and wind speed forecasting methods with different time horizons. In: North American Power Symposium, pp. 1–8 (2010)
49. Jung, J., Broadwater, R.P.: Current status and future advances for wind speed and power forecasting. Renew. Sustain. Energy Rev. **31**, 762–777 (2014)
50. Lawan, S.M., Abidin, W.A.W.Z., Chai, W.Y., Baharun, A., Masri, T.: Different models of wind speed prediction; a comprehensive review. Int. J. Sci. Eng. Res. **5**, 1760–1768 (2014)
51. Pohl, R., Kaltschmitt, M., Holländer, R.: Investigation of different operational strategies for the variable operation of a simple reverse osmosis unit. Desalination **249**, 1280–1287 (2009)
52. Zeng, G.Q., Xie, X.Q., Chen, M.R., Weng, J.: Adaptive population extremal optimization-based PID neural network for multivariable nonlinear control systems. Swarm Evol. Comput. 1–15 (2018)
53. Kang, J., Meng, W., Abraham, A., Liu, H.: An adaptive PID neural network for complex nonlinear system control. Neurocomputing **135**, 79–85 (2014)
54. Chen, J., Huang, T.C.: Applying neural networks to on-line updated PID controllers for nonlinear process control. J. Process Control **14**, 211–230 (2004)
55. Kumar, R., Srivastava, S., Gupta, J.R.P.: Artificial neural network based PID controller for online control of dynamical systems. In: IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), pp. 1–6 (2016)
56. González, R., Camacho, E., Montesinos, P., Rodríguez, J.A.: Optimisation of water demand forecasting by artificial intelligence with short data sets. Biosyst. Eng. (2018) (in press)
57. Herrera, M., Torgo, L., Izquierdo, J., Pérez-Garcá, R.: Predictive models for forecasting hourly urban water demand. J. Hydrol. **387**, 141–150 (2010)
58. Romano, M., Kapelan, Z.: Adaptive water demand forecasting for near real-time management of smart water distribution systems. Environ. Model. Softw. **60**, 265–276 (2014)

59. Tong, S., Lacap, J.H., Park, J.W.: Battery state of charge estimation using a load-classifying neural network. J. Energy Storage **7**, 236–243 (2016)
60. Xia, B., Cui, D., Sun, Z., Lao, Z., Zhang, R., Wang, W., Sun, W., Lai, W., Wang, M.: State of charge estimation of lithium-ion batteries using optimized Levenberg-Marquardt wavelet neural network. Energy **153**, 694–705 (2018)
61. Weigert, T., Tian, Q., Lian, K.: State of charge estimation of lithium-ion batteries using optimized Levenberg-Marquardt wavelet neural network. J. Power Sources **196**, 4061–4066 (2011)
62. Zhang, X., Wang, Y., Liu, C., Chen, Z.: A novel approach of battery pack state of health estimation using artificial intelligence optimization algorithm. J. Power Sources **376**, 191–199 (2018)
63. Chemali, E., Kollmever, P.J., Preindl, M., Emadi, A.: State-of-charge estimation of Li-ion batteries using deep neural networks: a machine learning approach **400**, 242–255 (2018)
64. Abdel-Khalik, A., Elserougi, A., Massoud, A., Ahmed, S.: A power control strategy for flywheel doubly-fed induction machine storage system using artificial neural network. Electr. Power Syst. Res. **96**, 267–276 (2013)
65. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Netw. **2**, 359–366 (1989)
66. Masters, T.: Practical Neural Network Recipes in C++. Morgan Kaufmann Publishers, California (1993)

Chapter 6
Control of Complex Biological Systems Utilizing the Neural Network Predictor

Samuel Oludare Bamgbose, Xiangfang Li and Lijun Qian

Abstract Intelligent control of complex systems faces many challenges including difficulty in realizing the model of the system and the need to address uncertainties. Because a lot of data are collected in modern systems, a data-driven approach can be employed to design intelligent control algorithms. Specifically, machine learning can be used to take advantage of the available datasets and predict the behavior of the system for improved design and performance of the controller. For example, in this chapter, a time-shifted neural network predictor is integrated with a proportional–integral controller to compensate for performance errors associated with time lag and nonlinear absorption pattern of meal and insulin in closed-loop blood glucose control systems. Additional benefits of this approach include the mitigation of errors that may be associated with sensor drift and slow change in concentration of the interstitial fluid glucose measured by the continuous glucose monitors. Different control approaches and devices for blood glucose control were reviewed, and simulation studies were presented to show the effectiveness of a neural network integrated control approach.

6.1 Introduction

In recent years, there has been notable development of control algorithms to solve problems associated with cardiovascular and endocrine systems. In the cardiovascular area, cardiac assist devices, which utilizes a mechanical pump has been developed to provide cardiac pressure support in order to achieve normal blood circulation in a patient's body. Baxter/Novacor left ventricular assist device (LVAD) was the first

S. O. Bamgbose (✉) · X. Li · L. Qian
Department of Electrical and Computer Engineering and CREDIT Center,
Prairie View A&M University, Prairie View, TX 77446, USA
e-mail: sobams77@yahoo.com

X. Li
e-mail: xili@pvamu.edu

L. Qian
e-mail: liqian@pvamu.edu

of such device to be approved by the Food and Drug Administration (FDA) in 1998. However, there are still control challenges to be addressed, which include adaptability to changing demands caused by varying physical activities, emotions, etc. Recent developments have explored real-time analysis, adaptive control, and model-based control [1, 2]. There are also control applications in the delivery of anesthesia, monitoring of oxygen saturation, implantable cardioverter defibrillator (ICD) and the intracardiac electrogram (IEGM) [3].

In the endocrine area, there is active research toward the realization of an artificial pancreas for diabetic patients. Diabetes is a disease that causes hyperglycemia (high blood glucose level) due to patient's difficulty in producing insulin—a hormone for converting glucose to energy which invariably regulates the blood glucose level (BGL). The goal of the United States National Science Foundation's (NSF) Smart and Connected Health Program is to develop solutions focused on well-being rather than disease [4]. Hence, in this chapter, a learning-based control system solution aimed at improving diabetic patient's well-being is presented.

Whereas Type 1 diabetes (T1D) is characterized by absolute lack of insulin, Type 2 (T2D) is characterized by insulin resistance and relative lack of insulin. Lifelong treatment is required by both T1D and T2D patients, which ranges from nutritional management and physical activities to intensive pharmacotherapy [5]. The goal of the intensive treatments is to optimally control the BGL in a way that hypo- or hyperglycemic episodes are eliminated or minimized. Hypoglycemia is the state of low BGL. When there is a sustained state of hyperglycemia, the patient's osmotic balance is altered leading to the dehydration of the body cells. As the glucose level increases further beyond the renal threshold, it changes the osmotic balance of the urine, such that fluids and electrolytes that normally would not be passed are released. In addition, tissue walls, including that of the kidneys, heart, eyes, and limbs could be destroyed. Moreover, there is high risk of death due to a dangerous dive of the patient's body pH or consumption of the tissue's protein. On the other hand, when hypoglycemia is sustained, there is a shortage of glucose needed by certain cells, especially in the brain and the retina, for metabolic processes. The inability of the cells to perform its metabolic functions could result in death.

The focus of the design in this chapter is on T1D, which can neither be prevented nor cured but can be treated effectively by external insulin infusion to regulate the BGL [6]. The insulin pump must mimic the mechanism by which the pancreas maintain appropriate BGL. However, a challenge in insulin therapy is the nonlinear insulin absorption pattern, how to tailor insulin regimen to individual patient's need, and accurate measurement of BGL [5, 7].

Since a lot of data are collected in modern systems, application of bio-inspired learning techniques such as neural network (NN) is an attractive approach to tackle such challenges. Although pharmacokinetic (PK) models have been developed to describe the elimination and absorption kinetics with the aim of understanding the glucose regulatory systems [8], those models have been criticized by reviewers for oversimplification of complex systems thereby not providing necessary accuracy for effective control studies [9]. Further, physiological models of glucose–insulin interaction with respect to different organs in the body have been developed [10, 11] but they have only been shown to work in a limited domain for some patients. In order

to help patients take countermeasures against impending hyper- or hypoglycemic periods, machine learning techniques have been employed to predict BGL excursions in [6] and [12]. The warning signal from the predictive model serves to trigger an insulin suspension or glucose intake action. Also, the authors of [13] and [14] used the AIDA simulator data to train a support vector machine (SVM) and recurrent neural network (RNN) models, respectively, for BGL prediction.

Other challenges restricting the effectiveness of a controller designed for BGL control are the limitations of the existing glucose sensors and the insulin pump. Whereas some glucose sensors are not implantable and inaccurate, others are implantable in the subcutaneous tissues but measure delayed values owing to the slow diffusion of glucose from the blood to the interstitial fluid [15]. In this chapter, an extended version of [16], a design integration between a conventional controller and a bio-inspired model is presented to mitigate the highlighted challenges associated with closed-loop BGL control for diabetic patients. Specifically, a time-shifted NN predictor was integrated with a proportional–integral controller (PI) to compensate for performance errors associated with time lag and nonlinear absorption pattern of meal and insulin, as well as sensor measurement inaccuracy. This framework will also allow researchers to investigate the effectiveness of control algorithms prior to clinical trials.

This chapter is organized as follows: Section 6.2 is an overview of the devices used in diabetes management, whereas Sect. 6.3 is a review of control algorithms employed for diabetes management. Section 6.4 discusses the concept of artificial pancreas and Sect. 6.5 presents the proposed design for blood glucose level control with neural network predictor for diabetic patients. Sect. 6.6 concludes the chapter.

6.2   Emergence of Devices in Diabetes Management

Multiple daily insulin injections or continuous subcutaneous insulin infusion had been employed to minimize hyperglycemia but early methods unfortunately increased the risk of hypoglycemia significantly. Kadish developed the first insulin pump in 1964 [17] and Young et al. made a Biostator, which was the first computerized insulin delivery device in the early 1980s [18]. However, early insulin pumps were impractical because they were too large, not precise and have other technical limitations. Several devices for measuring blood glucose (BG) and delivering insulin were developed between 1965 and 1980, and in 1983, the first commercial insulin pump was made [19]. Since then, modifications and improvements have been made with respect to size, insulin delivery method, features, and functionality. Advancements over the years have encouraged increased usage of the insulin pumps. However, for effective use, patients are required to have a basic understanding of insulin pharmacodynamics and carbohydrate bolus calculation. The burden of patients having to take finger-prick glucose measurements several times a day in order to adjust insulin doses led to the development of continuous glucose monitors (CGMs) in the late 1990s, which in turn has spurred efforts to connect it with the insulin pump in a closed-loop fashion [20].

The CGMs are more compact and more accurate than the old glucose meters. It has also proven to be more effective for glycemic control when used with insulin pumps. It consists of a sensor, a transmitter, and a receiver. The sensor is usually subcutaneously implanted to measure glucose level in the interstitial fluid, which makes it less invasive. The measurements are relayed to the receiver by the transmitter in real time. It could provide additional information including glucose values and trends, as well as alarms predicting hyper- or hypoglycemic episodes. One of the challenges with CGMs is their inaccuracy, with an average absolute error being 12.8 mg/dL or higher depending on the type of monitor. This is partly because the time lag before systemic glucose concentration change appears in the interstitial fluid has been estimated to be 4–26 min [21]. Even with good calibration, there can be 15–20% error [22], and the latest improvement has been marginal. Another limitation is the required frequent recalibration due to loss of sensitivity over time. Therefore, there is still the need for capillary blood glucose measurements using the fingerstick to correct any bias. Also, there could be lost or attenuated CGM signals.

Although the use of insulin pumps and CGMs reduces the patient's burden concerning BGL management, those devices typically work in open-loop fashion, and also still require patient's interpretation and manual compensation for metabolic disturbances. Furthermore, despite the smart features of the contemporary CGMs, existing BGL control is susceptible to errors associated with patient's inaction. That underscores the importance of closed-loop control of BGL for optimal replacement therapy.

6.3 Review of Control Algorithms for Diabetes Management

There have been several efforts aimed at applying control algorithms to blood glucose regulation using the pharmacokinetic (PK) models that describe the glucose–insulin dynamics in diabetic patients but those models do not provide accurate representation as they are oversimplified and are replete with assumptions that are unsuitable for real-life application. A fuzzy logic controller (FLC) with insulin pump in the loop for glucose level regulation of the Bergman model was proposed in [23]. The design was based on two input variables and one output variable, with a total of 49 IF-THEN rules defined. The error and error rate are the input variables while the insulin infusion rate is the output variable. The results obtained from using FLC was shown to be better than those obtained from using PID for corresponding patients. However, it was assumed that patients would not ingest meal for eight hours after the insulin infusion, BGL can be accurately measured, and the insulin absorption pattern was oversimplified. Hence, the performance of the controller to real patient may drastically deviate from the simulation result presented.

The authors of [24] proposed a model-based control strategy for blood glucose regulation of the Bergman patient model by calculating the optimal insulin delivery rate offline using parametric programming. The states of the model are as follows: the plasma glucose concentration above the basal value, the plasma insulin

concentration above the basal value, and a variable proportional to the plasma insulin in the remote compartment. The control variable is the insulin infusion rate, and the approach incorporates uncertainty in the patient model. Since the explicit delivery rate function was obtained offline, online computations during implementation are minimized. Nonetheless, assumptions for this strategy include widely spaced meal intake, oversimplified insulin absorption pattern and directly measurable BGL.

The simulation of a proportional–derivative (PD) and proportional–integral–derivative (PID) control of Hovorka patient model was presented in [25]. It was shown that the performance of the PID controller was better than that of the PD controller as the integral term eliminates the steady-state error. Insulin absorption pattern was included in the model, and five meals of varying boluses were considered. However, it took more than 12 h for the BGL to reach the target range.

A PID control strategy with insulin feedback to regulate blood glucose level was presented in [26]. Nine subjects already diagnosed with T1D, consisting of five males and four females with a median age of 44 years were recruited. Patients were asked to take measurements of their blood glucose and carbohydrate ingestion for 3 days. After an interval of at least 1 week, studies were conducted on the subjects for 30 h using the PID algorithm with insulin feedback. Supplemental carbohydrates were however needed to correct hypoglycemia due to PK delay relating to subcutaneous delivery of insulin.

An empirical algorithm for overnight blood glucose regulation based on hourly blood glucose measurement was proposed by [27]. Two overnight experiments were performed on 21 subjects on each of a 3 times visit at the study site, resulting in 138 overnight experiments. On each visit, patient's insulin therapy was administered on the first night and the control algorithm applied on the second night adopting venous BG measurements on an hourly basis. It was observed that regulation of BG improved from 52.9% on the first night to 72.2% on the second night. Also, interventions for hypoglycemic episodes reduced from 14 to 1. However, the patients' meal intake during the day was tightly controlled in order to minimize venous BGL excursion based on the insulin therapy.

6.4 The Concept of Artificial Pancreas

An artificial pancreas is a closed-loop system consisting of synthetic components working as a substitute for endocrine pancreas. But major technical problems in the development of a fully integrated closed-loop system include sensor drift, inaccuracy of the interstitial fluid glucose measurement taken by the CGM, the time lag and pattern of carbohydrate absorption, and the nonlinear peak insulin action with variability among patients. Regardless of the insulin delivery method, the pharmacodynamic action of even rapid-acting insulin has a long timescale. After insulin infusion, it takes about 90 min for the insulin to reach peak action, and its effect on the glucose continues for 6–8 h. Although the meal intake effect usually

peaks faster compared to the pharmacodynamic behavior of the insulin, its effect can continue for 3–8 h [28–30].

Closed-loop systems are differentiated by the control algorithm employed as well as the method of insulin delivery and glucose measurements. Whereas one control algorithm utilizes the proportional, integral and/or derivative components of the physiological characteristics, others are predictive in nature. Different types of glucose sensing and insulin delivery include subcutaneous (SC) sensing and SC delivery, intravenous (IV) sensing and IV delivery, and IV sensing and intraperitoneal delivery. The first mode of sensing and delivery is the most studied as the invasion is minimal, although there is a time lag in glucose measurements. The other modes require surgery and still have physiological delays, albeit comparatively reduced overall delay [5].

The authors of [31] studied the feasibility of the Medtronic MiniMed external physiological insulin delivery system in youth with T1D. They concluded that overnight closed-loop control performance was better than day time due to no meal disturbance and proposed an additional premeal priming bolus of insulin to improve postprandial glycemic excursions caused by peak plasma insulin action occurring 1–2 h after insulin delivery. The mean relative absolute deviation of the sensor from the venous blood glucose was 13.2 +/− 10.9%, and there is concern relating to the risk of hypoglycemia.

6.5 A Closed-Loop Blood Glucose Control Design with Neural Network Predictor for Diabetic Patients

Despite the recent advancements in glycemic control for diabetic patients, the realization of an automated closed-loop artificial pancreas is still a challenge. The design presented in this section is an integrated control system for in silico closed-loop administration of insulin for T1D patients based on patients' medical record and real-time control-relevant data [32]. The proposed system consists of a virtual patient model, a neural network predictor trained on patients' data for feedback purposes, a PI Controller and data logging nodes. The virtual patient takes into account the delayed and time-varying insulin and carbohydrate absorption rate associated with the existing subcutaneous insulin delivery and complex glucose metabolism, respectively. The neural network predictor was trained using 23 features including semi-static and dynamic data, with built-in knowledge of all available past blood glucose levels. Then the controller calculates the infusion bolus to be delivered by the insulin pump. Extensive simulations are performed, and it is shown that the proposed data-driven closed-loop system for glycemic control can effectively regulate the blood glucose level of T1D patients without hyper- or hypoglycemic excursions, and with no preset instruction on meal ingestion. Moreover, the neural network predictor has less root-mean-square error (RMSE) compared with the currently used CGMs, which takes measurement from the interstitial fluid.

6.5.1 System Architecture

In this subsection, the structure of the integrated closed-loop control system for automatic insulin administration is described. It is made up of a virtual patient [33], a time-shifted neural network predictor, data logging nodes, and a PI controller to compute appropriate insulin boluses by the insulin pump as described in Figure 6.1.

The target BGL is denoted by g, whereas $\hat{g}$ denotes the BGL observation from the NN predictor, and u is the insulin infusion command. The NN predictor takes patients' semi-static data, as well as dynamic data and command signal from the insulin pump to compute BGL that will be reached at a future time for feedback purpose. This framework takes into account the delayed, continuous, and time-varying action of the insulin associated with the subcutaneous insulin delivery route and also provides a means to not only obtain better observation than the existing glucose monitors but also future measurements.

6.5.2 Blood Glucose Prediction

The free online AIDA diabetes simulator [34] was used to generate data which was prepared with built-in past BGL and Insulin information. Data setup, neural network training, and performance evaluation are described in the following subsections.

6.5.2.1 Data Setup

The training data have 23 features that can be categorized as semi-static and dynamic for a prediction window of 9 h. The semi-static data are the weight, renal threshold of glucose, creatinine clearance rate, hepatic insulin sensitivity, peripheral insulin sensitivity, initial plasma insulin level, and initial blood glucose level, which are denoted as wt, rtg, ccr, sh, sp, pb, and $g0$, respectively. The dynamic data are the
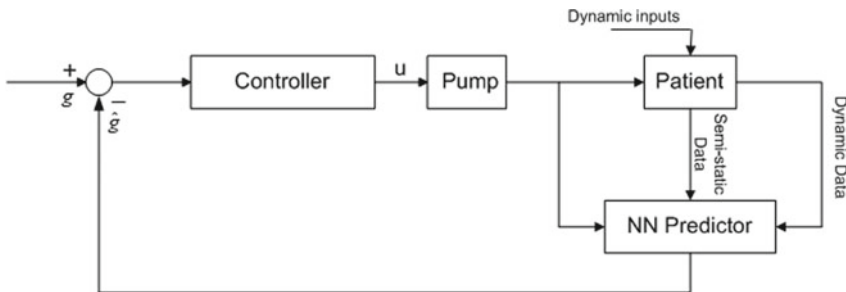
Fig. 6.1 Data-driven blood glucose control system

sampling time, sampled blood glucose levels through the prediction window, up to three carbohydrate intake along with ingestion time within the prediction horizon, and the four infusion boluses by the insulin pump, which are denoted as t_s, $g1$, $g2$, $g3$, $g4$, $g5$, $m1$, $m1t$, $m2$, $m2t$, $m3$, $m3t$, $u1$, $u2$, $u3$, and $u4$. There were 2,100 sample data generated, which represent 300 patient blood glucose profiles for 9 h simulation. Knowledge of BGL history was built into the data as part of the features to enhance prediction performance. Each dynamic input vector precedes the current time step. A sampling rate of 90 min was selected to compensate for the delayed and time-varying subcutaneously injected insulin action, which peaks between 1 and 2 h. A shorter rate could lead to hypoglycemia as insulin is administered before the last infusion could take effect. A larger rate was not selected so that glucose absorption and increase due to meal ingestion, which could be multiple, can be counteracted in a timely fashion. Four regular-type insulin infusion were applied at the specified sampling rate, and up to three random-sized meals can be ingested.

6.5.2.2 Predictor Model and Training

The complex nature of glucose metabolism and insulin delivery, as well as data-intensive management of diabetes, makes machine learning models attractive for describing hidden processes. Neural networks are nonlinear mapping models consisting of processing units called neurons, which interact with other neurons through weighted connections as described in [35] and shown in Figure 6.2.

As shown below, the neurons are structured into two or more layers after the input layer: the hidden and output layers. The process of extracting knowledge or information based on the structure is called training or learning. In supervised learning applied in this work, each data input sample (with multiple features) have an output label, and a portion of that dataset is used to adjust the weight of the NN by minimizing an error function. The remaining portion of the dataset is used to test the performance of the trained network.
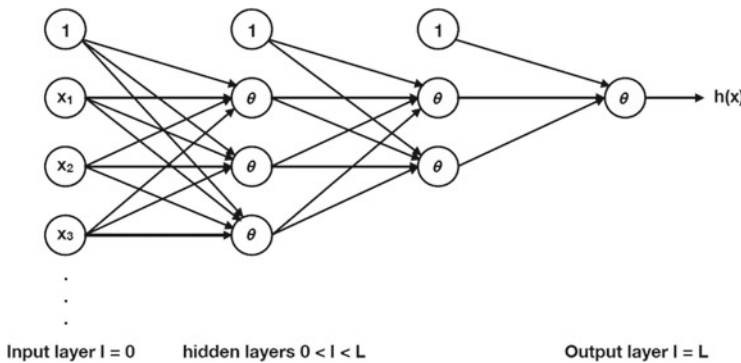
Fig. 6.2 Feedforward neural network structure

The feedforward neural network model described by (6.1) and (6.2) was employed but with knowledge of past predictor outputs built into the input data as described in the previous subsection.

$$\hat{g}_n^{(l+1)}(k) = \theta(s_n^{(l+1)})(k) \tag{6.1}$$

$$s_n^{(l+1)}(k) = \mathbf{w}_n^{(l+1)}(k)\hat{\mathbf{g}}^l(k) + b_n^{(l+1)}(k) \tag{6.2}$$

where l denotes layer, n denotes unit or neuron, θ is an activation function, and k is the time step. $\hat{\mathbf{g}}^l$ is the output vector from layer l ($\hat{\mathbf{g}}^0$ is the original input vector), $\mathbf{s}^l$ is the input vector into layer l, w_n^l are the weights from layer $l-1$ to unit n of layer l, and b^l are the biases from layer $l-1$ to unit n of layer l.

A two-layer neural network model was trained with eight hidden neurons using the NN toolbox in MATLAB. Considering the heuristic that the number of sample data should be ten times larger than the weight dimension and experimenting with a different number of hidden neurons, it was observed that eight hidden neurons was optimal for the considered application. Furthermore, the built-in past BGL knowledge in the data setup provides unique performance improvement for the presented application. Levenberg–Marquardt algorithm [36–38] was used for training, which is considered a faster algorithm than the standard back-propagation algorithm, and data splitting into 70% training, 15% validation, and 15% testing was performed by "dividerand" function.

6.5.2.3 Performance Evaluation

The predictor performance was measured using the RMSE, and the output versus target regression coefficient (R). The goal is to obtain an RMSE that is closer to zero relative to the magnitude of the predicted values and a correlation coefficient that is closer to 1. As shown in Figure 6.3, the training, validation, and testing mean square error (MSE) were 20, 29, and 35, respectively. Hence, the RMSEs were 4.5, 5.4, and 5.9 mg/dL, respectively, whereas the currently used continuous glucose monitor deviates by 15–20% from the actual blood glucose values [22]. Also, due to the similar characteristics of the validation and test error curves, there is no significant overfitting. The regression plots in Figure 6.4 showed good fits between the predicted outputs and the targets with high R values.

6.5.2.4 Justification for Using the AIDA simulator

The reasons for adopting the AIDA online simulator [34] otherwise called the virtual patient in generating patient's data are: (1) Real patients' data are difficult to obtain due to privacy and ethical issues. (2) Experiments on human subject are costly and time-consuming. (3) Large dataset can be generated by the simulator. (4) Greater
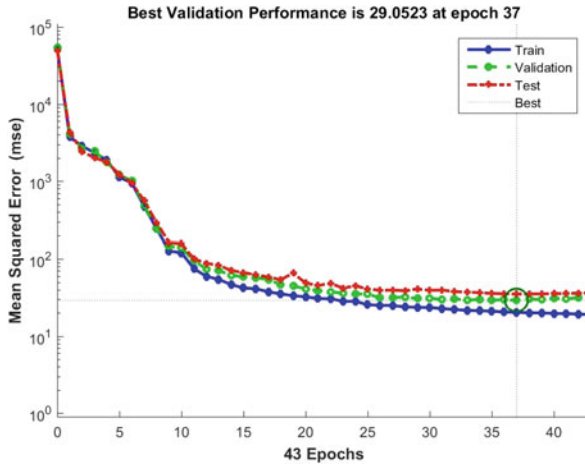
Fig. 6.3 Neural network predictor performance
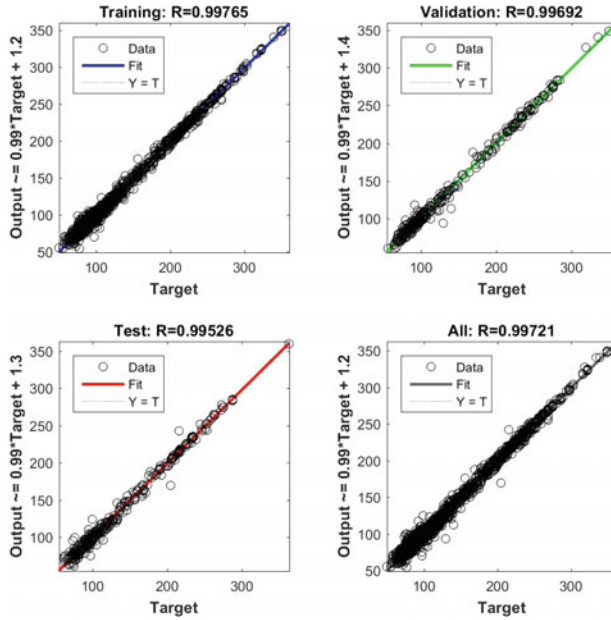
Fig. 6.4 Neural network predictor output versus target regression

flexibility can be achieved as specific scenarios can be simulated. Furthermore, the British Diabetic Association (BDA) conducted an independent assessment based on feedback from internal assessors as well as healthcare professionals. Following a fairly accurate rating by healthcare professionals, the BDA decided to catalogue the simulator in the BDA's healthcare professional brochure [39]. More details about the simulator model and limitations such as non-capture of stress, exercise, alcohol, etc., are reported in [40, 41].

6.5.3 Glycemic Control Design and Insulin Delivery

A PI controller was designed to compute the control command for the appropriate insulin dose at predefined discrete times. This controller is suiting to our application as it does not require the mathematical description of the complex physiological processes relating to BGL in the body to compute the control command. The insulin infusion command is a function of the difference between the target glucose level and the observation as described in (6.3).

$$u(k) = \begin{cases} K_p e(k) & \text{if } k = 0 \\ K_p\left[e(\hat{k}) + \dfrac{\Delta t}{\tau_I}(e(0) + \sum_{i=1}^{k} e(\hat{i}))\right] & \text{otherwise} \end{cases} \quad (6.3)$$

$$e(k) = g - \hat{g}(k) \quad (6.4)$$

$$e(\hat{k}) = g - \hat{g}(k + \tau) \quad (6.5)$$

The proportional gain (K_p) and the integral gain ($\dfrac{K_p}{\tau_I}$) are the tunable parameters. Δt is the sample time, k is the current time step, and τ is the positive time shift. BGL observation was time-shifted by 30 min to capture the long-term effect of the time-varying insulin action due to nonlinear subcutaneous insulin delivery for a more effective control action. Hence, $e(k)$ is the error between the BGL target and predictor output at the current time step, whereas $e(\hat{k})$ is the error between the BGL target and predictor output at a specified future time. The initial BGL observation was not time-shifted as there was prior infusion, and invariably no active insulin action. The proportional gain adjusts the insulin delivery with respect to the error signal, while the integral gain adjusts insulin delivery with respect to the sum of all past errors. The derivative term was not used as the rate of change of BGL over time fluctuates with meal disturbance and insulin infusion.

The goal of the controller is to maintain the BGL within 70 and 140 mg/dL two hours after meal ingestion as typical for non-diabetic patients in [42]. The design was done using MATLAB toolbox. By tuning the control parameters, the optimal performance was obtained with $K_p = -0.078$ and $\frac{K_p}{\tau_I} = -0.00015$. The predictor,
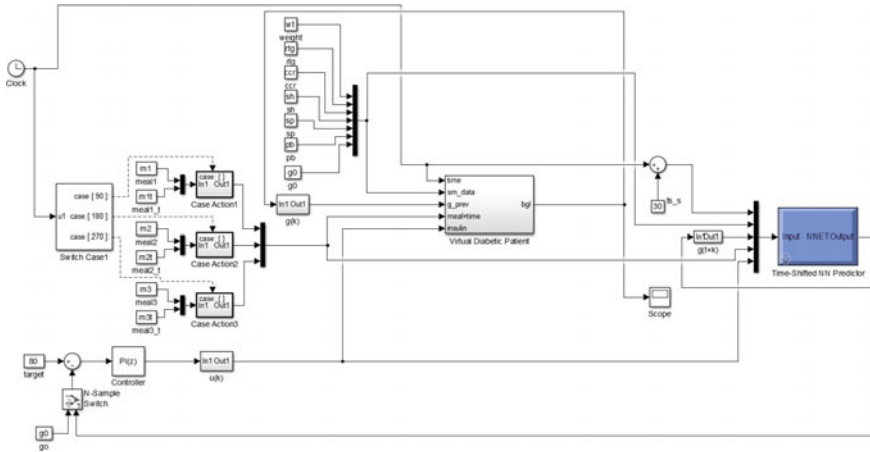
Fig. 6.5 Intelligent blood glucose control system model

controller, and sensor nodes were integrated as shown in Figure 6.5. The system was implemented in silico using MATLAB Simulink with predictor outputs having RMSE of 5.9 mg/dL relative to the virtual patient. Existing subcutaneous insulin delivery mode as well as random meal intake pattern was considered in this work. Therefore, the existing insulin pumps can be easily utilized and patients have the freedom to embrace any meal pattern of their choice. The integrated system was able to dynamically and automatically set insulin infusion without hyper- or hypoglycemic excursions.

6.5.4 Simulation Results

In order to prove the effectiveness and robustness of our system, the controller performance for five patients with diverse medical details and meal ingestion patterns were simulated as follows. The variables were selected in a way to present diverse situation that the system may encounter in practice as shown in Table 6.1, where wt is the weight in lb; rtg is the renal threshold of glucose in mg/dL; ccr is the creatinine clearance rate in mL/min; sh is the hepatic insulin sensitivity, sp is the peripheral insulin sensitivity; m1, m2, and m3 are the meal carbohydrate contents in g; and m1t, m2t, and m3t are the meal ingestion times in min.

Case I represents a patient with relatively large weight and large carbohydrate ingestion pattern, whereas Case II is a patient with medium weight and medium carbohydrate ingestion pattern. A patient with small weight and small carbohydrate ingestion was described by Case III while Case IV depicts a patient with large weight and medium carbohydrate ingestion. Finally, Case V characterize a patient with small

Table 6.1 Patients' semi-static data and meal ingestion pattern

	wt	rtg	ccr	sh	sp	m1	m1t	m2	m2t	m3	m3t
Case 1	191	176	120	0.8	0.8	39	30	57	120	63	245
Case 2	152	160	90	0.5	0.7	35	32	43	120	55	255
Case 3	128	121	82	0.3	0.8	28	50	23	135	39	240
Case 4	240	192	120	0.5	0.5	36	37	29	155	43	252
Case 5	101	150	100	0.7	0.8	41	40	53	142	67	265

The acronyms are defined in Subsubsection 6.5.2.1

weight but large carbohydrate ingestion. Other essential patient vitals are as provided in Table 6.1.

Figure 6.6 showed that the control system was able to keep the blood glucose level between 70 and 140 mg/dL two hours after meal ingestion, which is consistent with the standard of American Diabetic Association (ADA) [42] for non-diabetic patients. The control system achieved the goal of normo-glycemia till the end of the simulation without hyper-or-hypoglycemic excursions in different cases that may be encountered in practice as shown in sub figures a–e, despite varying patients' medical data and random meal intake pattern. This preclinical simulation setup yields results in a fraction of time required for clinical trials and can help to guide clinical experiments.

6.6 Conclusion

Pharmacokinetic models of diabetic patients utilize a theoretical number of compartments to describe elimination and absorption kinetics which does not provide enough accuracy for effective control studies. A machine learning-based approach to designing an intelligent controller has been presented in this chapter. Specifically, a neural network predictor has been trained to describe the complex glucose–insulin relationships for Type 1 diabetic patients based on virtual patient's data. Outputs from the time-shifted predictor were fed back to the controller to compute insulin boluses for the virtual patient.

The simulation results showed that the designed control system can effectively administer insulin automatically to regulate blood glucose level within normal range, irrespective of the patient's meal intake pattern. Furthermore, the predictor performance was shown to be better than that of implanted sensors which are affected by the body's immune response and delayed diffusion of glucose from the blood to the subcutaneous tissue. The proposed simulation framework is a time- and cost-effective tool for guiding clinical studies toward the development of artificial pancreas. The presented approach can be extended to Type 2 diabetes, and it is expected that even better results can be obtained by utilizing more data samples. Further research will require real-patient data and consideration of other factors that may influence blood glucose level such as stress, exercise, etc., as well as in vivo testing of the control strategy.
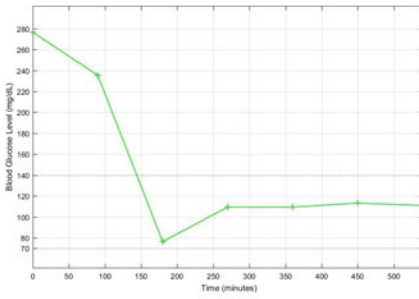
(a) Case I
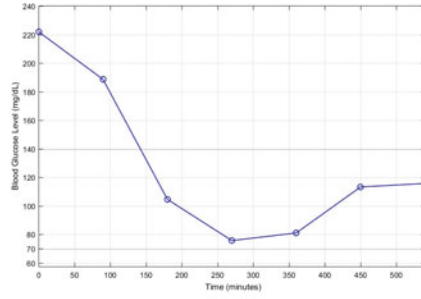
(b) Case II
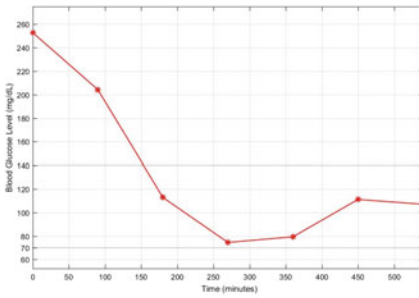
(c) Case III
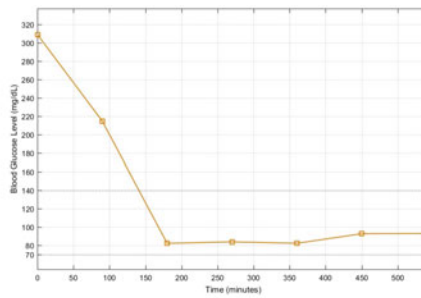
(d) Case IV
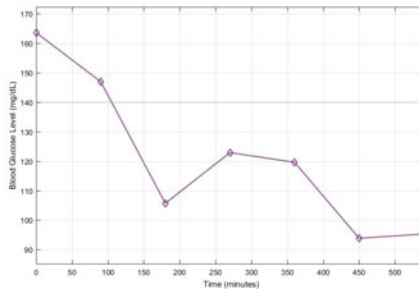
(e) Case V

Fig. 6.6 Control system performance for five diverse virtual patients

References

1. Haddad, S.A.P., Houben, R.P.M., Serdijn, W.A.: The evolution of pacemakers. Eng. Med. Biol. Mag. **25**, 35–39 (2006)
2. Schima, H., Vollkron, M., Jantsch, U., et al.: First clinical experience with an automatic control system for rotary blood pumps during ergometry and right-heart catheterization. J. Heart Lung Transplant. **25**, 167–173 (2006)
3. Doyle, J.F., Bequette, B.W., Middleton, R., et al.: Control in biological systems. In: Smad, T., Annaswamy (eds.) The Impact of Control Technology. IEEE Control Systems Society (2011). https://www.epfl.ch/labs/la/wp-content/uploads/2018/08/IoCT-FullReport.pdf. Cited 25 Sep 2018
4. Smart and Connected Health (SCH): National Science Foundation (2016). https://www.nsf.gov/pubs/2013/nsf13543/nsf13543.htm. Cited 13 Dec 2016
5. Penfornis, A., Personeni, E., Borot, S.: Evolution of devices in diabetes management. Diabetes Technol. Ther. (2011). https://doi.org/10.1089/dia.2011.0058
6. Plis, K., Bunescu, R., Marling, C., et al.: A machine learning approach to predicting blood glucose levels for diabetes management. In: Twenty-Eighth AAAI Conference on Artificial Intelligence, pp. 35–39 (2014)
7. Wilinska, M.E., et al.: Insulin kinetics in type-1 diabetes: continous and bolus delivery of rapid acting insulin. IEEE Trans. Biomed. Eng. (2005). https://doi.org/10.1109/TBME.2004.839639
8. Holz, M., Fahr, A.: Compartment modeling. Adv. Drug Deliv. Rev. (2001). https://doi.org/10.1016/S0169-409X(01)00118-1
9. Farmer, T.G., Edgar, T.F., Peppas, N.A.: The future of open and closed-loop insulin delivery systems. J. Pharm. Pharmacol. (2008). https://doi.org/10.1211/jpp.60.1.0001
10. Lehmann, E.D., Deutsch, T.: A physiological model of glucose-insulin interaction in type 1 diabetes mellitus. J. Biomed. Eng., 235–242 (1992)
11. Wilinska, M.E., Chassin, L.J., Acerini, C.L., et al.: Simulation environment to evaluate closed-loop insulin delivery systems in type 1 diabetes. J. Diabetes Sci. Technol. (2010). https://doi.org/10.1177/193229681000400117
12. Sudharsan, B., Peeples, M.: Hypoglycemia prediction using machine learning models for patients with type 2 diabetes. J. Diabetes Sci. Technol. (2015). https://doi.org/10.1177/1932296814554260
13. Reymann, M.P., Dorschky, E., Groh, B.H., et al.: Blood glucose level prediction based on support vector regression using mobile platforms. In: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 2990–2993 (2016)
14. Robertson, G., Lehmann, E., Sandham, W., et al.: Blood glucose prediction using artificial neural networks trained with the AIDA diabetes simulator: a proof-of-concept pilot study. J. Electr. Comput. Eng. (2011). https://doi.org/10.1155/2011/681786
15. Leelarathna, L., et al.: Evaluating the accuracy and large inaccuracy of two continous glucose monitoring systems. Diabetes Technol. Ther. (2013). https://doi.org/10.1089/dia.2012.0245
16. Bamgbose, S.O., Li, X., Qian, L.: Closed loop control of blood glucose level with neural network predictor for diabetic patients. In: 2017 IEEE 19th International Conference One-Health Networking, Applications and Services (Healthcom), Dalian, pp. 1–6 (2017). https://doi.org/10.1109/HealthCom.2017.8210817
17. Kadish, A.H.: A servomachanism for blood sugar control. Biomed. Sci. Instrum. **1**, 171–176 (1963)
18. Young, A., Herf, S.: Biostator glucose controller: a building block of the future. Dibetes Educ. **10**, 11–12 (1984)
19. Shah, V.N., Shoskes, A., Tawfik, T.: Closed-loop system in the management of diabetes: past, present, and future. Diabetes Technol. Ther. (2014). https://doi.org/10.1089/dia.2014.0193
20. Garg, S.K., Hirsch, I.B.: Self-monitoring of blood glucose—an overview. Diabetes Technol. Ther. (2014). https://doi.org/10.1089/dia.2014.1501
21. Pandit, K.: Continuous glucose monitoring. Indian J. Endocrin. Metabol. (2012). https://doi.org/10.4103/2230-8210.104056

22. Griffin, S., Ballard, D.: Continuous glucose monitoring (2017). http://2aida.net/welcome/.Cited 08 May 2017
23. Maleki, A., Geramipour, A.: Continuous control of blood glucose in TIDM using fuzzy logic controller in insulin pump: a simulation study. In: The 2nd International Conference on Control, Instrumentation and Automation, pp. 122–127 (2011)
24. Dua, P., Doyle, F.J., Pistikopoulos, E.N.: Model-based blood glucose control for type 1 diabetes via parametric programming. IEEE Trans. Biomed. Eng. (2006). https://doi.org/10.1109/TBME.2006.878075
25. Mahmud, F., Isse, N., Daud, N.: Evaluation of PD/PID controller for insulin control on blood glucose regulation in a type 1 diabetes. In: International Conference on Engineering, Science and Nanotechnology, pp. 030072-1–030072-7 (2016)
26. Steil, G., et al.: Effect of insulin feedback on closed loop glucose control. J. Clin. Endocrinol. Metab. (2011). https://doi.org/10.1210/jc.2010-2578
27. Patte, C., et al.: Feasibility of overnight closed-loop control based on hourly blood glucose measurements. J. Diabetes Sci. Technol. (2012). https://doi.org/10.1177/1932296814554260
28. Klonoff, D.: The artificial pancreas: how sweet engineering will solve bitter problems. J. Diabetes Sci. Technol. **1**, 72–81 (2007)
29. Bequette, B.W.: Challenges and progress in the development of a closed-loop artificial pancreas. In: 2012 American Control Conference (ACC), pp. 4065–4071 (2012)
30. Thabit, H., Hovorka, R.: Closed-loop insulin delivery in type 1 diabetes. Endocrinol. Metab. Clin. North. Am. (2012). https://doi.org/10.1016/j.ecl.2011.12.003
31. Weinzimer, S., et al.: Fully automated closed-loop insulin delivery versus semiautomated hybrid control in pediatric patients with type 1 diabetes using an artificial pancreas. Diabetes Care (2008). https://doi.org/10.2337/dc07-1967
32. Bamgbose, S.O., Li, X., Qian, L.: Closed loop control of blood glucose level with neural network predictor for diabetic patients. In: 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom), pp. 1–6 (2017)
33. Lehmann, E.D., Deutsch, T.: AIDA technical guide (1996). http://2aida.net/welcome/. Cited 30 Oct 2016
34. Lehmann, E.D., et al.: On-line simulation (1996). http://2aida.net/welcome/. Cited 30 Oct 2016
35. Abu-Mostafa, Y.S., Magdon-Ismail, M., Lin, H.: Learning from Data. AMLbook.com, USA (2012)
36. Levenberg, K.: A method for the solution of certain problems in least squares. Quart. Appl. Math. **2**, 164–168 (1944)
37. Marquardt, D.: An algorithm for least-squares estimation of nonlinear parameters. SIAM J. Appl. Math. **11**, 431–441 (1963)
38. Ranganathan, A.: The Levenberg-Marquardt algorithm (2004). http://ananth.in/docs/Imtut.pdf. Cited 1 Sep 2004
39. Lehmann, E.D.: British diabetic association review of the AIDA v4 diabetes software simulator program. Diabetes Technol. Ther. **6**, 87–96 (2004)
40. Lehmann, E.D., Tarin, C., Bondia, J., et al.: Development of AIDA v4.3b diabetes simulator: technical upgrade to support incorporation of lispro, aspart, and glargine insulin analogues. J. Electr. Comput. Eng. (2011). https://doi.org/10.1155/2011/427196
41. Reed, K., Lehmann, E.D.: Interactive educational diabetes/insulin tutorial at www.2aida.info. Diabetes Technol. Ther. **8**, 87–96 (2006)
42. American Diabetes Association: Standards of medical care in diabetes—2017. Diabetes Care **40**, S1–S135 (2017)

Chapter 7
A Real-Time Big Data Control-Theoretical Framework for Cyber-Physical-Human Systems

Azwirman Gusrialdi, Ying Xu, Zhihua Qu and Marwan A. Simaan

Abstract Cyber-physical-human systems naturally arise from interdependent infrastructure systems and smart connected communities. Such applications require ubiquitous information sensing and processing, intelligent machine-to-machine communication for a seamless coordination, as well as intelligent interactions between humans and machines. This chapter presents a control-theoretical framework to model heterogeneous physical dynamic systems, information and communication, as well as cooperative controls and/or distributed optimization of such interconnected systems. It is shown that efficient analytical and computational algorithms can be modularly designed and hierarchically implemented to operate and optimize cyber-physical-human systems, first to quantify individually the input–output relationship of nonlinear dynamic behaviors of every physical subsystem, then to coordinate locally both cyber-physical interactions of neighboring agents as well as physical-human interactions, and finally to dynamically model and optimize the overall networked system. The hierarchical structure makes the overall optimization and control problem scalable and solvable. Moreover, the three levels integrate individual designs and optimization, distributed cooperative optimization, and decision-making through real-time, data-driven, model-based learning and control. Specifically, one of the contributions of the chapter is to demonstrate how the combination of dissipativity theory and cooperative control serves as a natural framework and promising tools to analyze, optimize, and control such large-scale system. Application to digital power grid is investigated as an illustrative example.

A. Gusrialdi (✉)
Tampere University, Tampere 33014, Finland
e-mail: azwirman.gusrialdi@tuni.fi

Y. Xu · Z. Qu · M. A. Simaan
University of Central Florida, Orlando 32816, USA
e-mail: ying.xu@ucf.edu

Z. Qu
e-mail: qu@ucf.edu

M. A. Simaan
e-mail: simaan@eecs.ucf.edu

7.1 Introduction

Cyber-physical-systems (CPSs) refer to the integrations of cyber core consisting of communication network, computation and physical processes (engineered systems) which are normally large scale and complex, as illustrated in Fig. 7.1. These two components are tightly coupled: embedded computers and networks monitor and control the physical processes, usually with feedback loops where physical processes affect computations and vice versa. In addition, CPSs will also interact with humans resulting in cyber-physical-human systems. Cyber-physical-human systems naturally arise from interdependent infrastructure systems and smart connected communities. Examples include smart grid [54], intelligent transportation systems [16], and smart city [5]. Such applications require ubiquitous information sensing and processing, intelligent machine-to-machine communication, a seamless coordination of physical systems, and intelligent interactions between humans and machines. While technological advances and the development of relatively inexpensive yet powerful communication, computation, and sensing devices make the realization of such complex system feasible, fundamental technical challenges centered on real-time big data processing, optimization, and control of the spatially distributed complex systems remain to be solved. A major and fundamental challenge is to develop a control design theory that does not consider the physical and cyber components separately, but as two facets of the same system [2]. Another major challenge is the choice of control architecture which allows the designer to control the complex system efficiently and in real time. Traditional centralized control architecture, where all the data from ubiquitous sensors are gathered in a centralized processing center, which optimizes and computes the control input for the overall system is not appropriate to optimize and control such large-scale interconnected system since it may suffer from explosion of data and may also harm data privacy [4]. This calls for a scalable and modular system theoretic tools to analyze, optimize, and control the cyber-physical-human systems. In particular, distributed optimization and control algorithms are highly desirable for dealing with such complex systems due to its scalability and robustness against component faults and cyberattacks [17].

The chapter presents a control-theoretical framework to model heterogeneous physical dynamic systems, information and communication, as well as cooperative controls and/or distributed optimization through which human operator or users can interact effectively with physical systems in a multi-agent setting to achieve various control and optimization objectives. It is shown that efficient computational algorithms can be applied hierarchically to operate and optimize cyber-physical-human systems, first individually to quantify the dynamic behavior of every agent, then locally to describe the local interactions of neighboring agents, and finally to the overall system. All the three control levels deal with real-time big data, and the hierarchical structure makes the overall optimization and control problem scalable and solvable. In particular, one of the contributions is to demonstrate how the concept of dissipativity theory and cooperative control serve as a natural framework
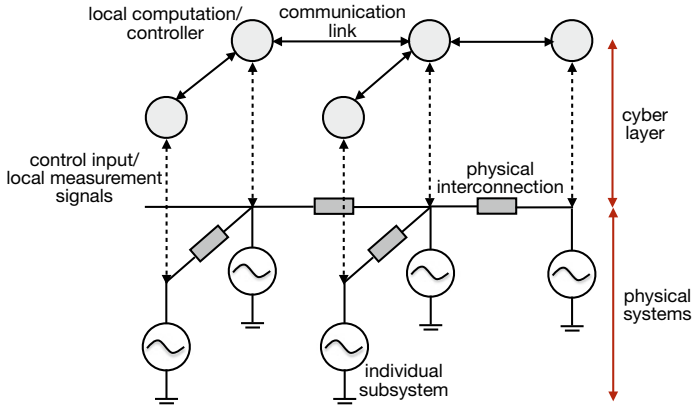
Fig. 7.1 An illustrative diagram of cyber-physical systems as exemplified by power system

and promising tools to analyze, optimize, and control such large-scale systems in a scalable and modular manner. Application to digital power grid is investigated as an illustrative example.

The chapter is organized as follows. We begin with dynamic modeling of cyber-physical-human systems together with its optimization and control objectives in Section 7.2. A brief summary of the basic concepts of dissipativity theory and cooperative control as the main analytical and design tools is presented in Section 7.3. Section 7.4 provides an example of applying the dissipativity theory and cooperative control to design hierarchical control of power system. Modeling and analysis of human–machine interaction with focus on electricity market are presented in Section 7.5. The role of real-time big data and decision-making in controlling cyber-physical-human systems is discussed in Section 7.6. Finally, we conclude in Section 7.7.

7.2 Dynamic Modeling of Cyber-Physical Systems and Its Optimization/Control Objectives

System modeling is an important step in designing control algorithms. Briefly speaking, a model is a mathematical representation of physical system which allows us to reason and predict how the system will behave. In this chapter, we are mainly interested in models of dynamical system describing the input/output behavior of systems. To this end, let us consider cyber-physical-human systems consisting of n heterogeneous physical systems whose individual dynamics can be modeled by differential equations in the form of

$$\dot{x}_i = f_i(x_i, u_i, r_i), \quad y_i = h_i(x_i, r_i), \tag{7.1}$$

with $i = \{1, \ldots, n\}$. The model in (7.1) is known as state-space models where variables $x_i \in \Re^{n_i}$ denote the state which encodes what needs to be known about the past history, $u_i \in \Re^m$ is the control signals to be designed, and $y_i \in \Re^m$ denotes the output (measurement) signals of the i-th system. In addition, $r_i \in \Re^m$ in (7.1) is the operational decision as a result of the intelligent interaction between humans and the physical systems which may take place in a slower timescale. In general, the physical systems may also be interconnected through a physical network whose characteristic could be described by the following algebraic equation:

$$\kappa_i(y_1, \ldots, y_n, x_1, \ldots, x_n) = 0. \tag{7.2}$$

As an example, consider a power system where the individual physical system refers to the synchronous generator as shown in Fig. 7.1. For the sake of simplicity, the dynamics of synchronous generator is given by the following swing equation:

$$M_i \ddot{\delta}_i = P_{m,i} - P_{e,i} - D_i \omega_0 \dot{\delta}_i, \tag{7.3}$$

where $M_i > 0$ denotes its inertia, $D_i > 0$ is its damping constant, $P_{m,i}$ denotes its mechanical power while $P_{e,i}$ is its active power output, and δ_i denotes its rotor angle measured with respect to a rotating frame with speed ω_0. The generators are physically interconnected with each other which can be characterized through the following nonlinear power flow equation:

$$P_{e,i} = E_i^2 G_{ii} + \sum_{k \neq i} E_i E_k (G_{ik} \cos \delta_{ik} + B_{ik} \sin \delta_{ik}), \tag{7.4}$$

where $\delta_{ik} = \delta_i - \delta_k$, E_i is the voltage of the generator bus, and $Y_{ik} = G_{ik} + jB_{ik}$ is the transfer admittance between generators i and k. Defining, respectively, the states, input and output of the i-th generator as $x_i = [\delta_i - \delta_i^*, \omega_i]^T$, $u_i = P_{m,i}$ and $y_i = x_i$ with δ_i^* denotes the final angle, we can recast swing equation (7.3) together with power flow equation (7.4) with respect to their equilibrium in the form of (7.1) as [22]

$$\dot{x}_i = A_i(x_i)x_i + B_i(x_i)u_i + \sum_{k \in \mathcal{N}_i} H_{ik}(y_i, y_k)(y_k - y_i), \quad y_i = C_i x_i, \tag{7.5}$$

where $\mathcal{N}_i$ denotes the neighboring set of generator i, matrices A_i, B_i and coupling matrix H_{ik} are state/output-dependent. Note that generators with higher (e.g., fifth or sixth) order dynamics can also be represented by state-space model (7.5). In addition to the physical network, there is also a cyber-layer representing information/communication network for the system operator/local controller of physical systems to obtain/exchange measurements in order to monitor and control the overall system. The structure of communication network (information flow) in general is modeled using a graph as illustrated in Fig. 7.1. Let $\mathcal{N}_i^c$ denote the communication neighboring set of the i-th subsystem. In other words, subsystem $j \in \mathcal{N}_i^c$ if infor-
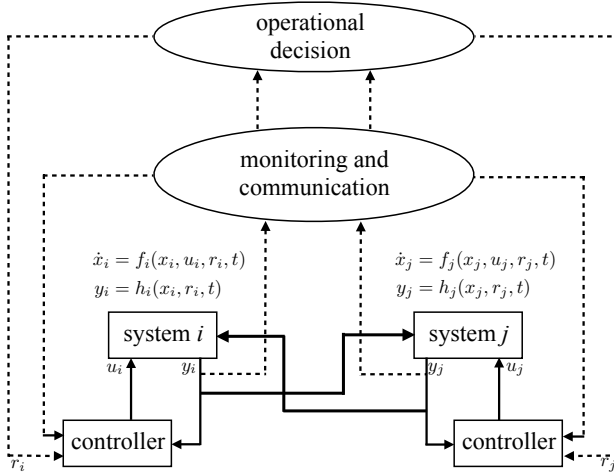
Fig. 7.2 Three-level data-driven controls of cyber-physical-human systems. The dashed lines represent information flow between different levels

mation on measurement y_j is available to the i-th subsystem. The communication network topology can also be represented by the following communication matrix:

$$S^c = [S^c_{ij}] \in \Re^{n \times n}, \quad S^c_{ii} = 1, \tag{7.6}$$

where $S^c_{ij} = 1$ if $j \in \mathcal{N}^c_i$ and $S^c_{ij} = 0$ otherwise.

Optimizing and controlling the above cyber-physical-human systems calls for computationally efficient and scalable algorithms to deal with its large-scale nature and complexity (in terms of heterogeneous individual nonlinear dynamics and their physical interconnections). To this end, we divide the control objective of cyber-physical-human systems into three levels as illustrated in Fig. 7.2. Specifically, the control input u_i in (7.1) is decomposed into the following hierarchical form:

$$u_i = u_{s_i}(x_i) + \underbrace{u_{l_i}(y_i, y_j) + v_i}_{\overline{u}_i}, \tag{7.7}$$

each layer with the following control design objective:

1. the lowest level control u_{s_i} aims to stabilize each individual physical system,
2. the mid-level control input u_{l_i} is to achieve a local coordination for a group of physical systems, and
3. the highest level control v_i aims at ensuring stability of the overall interconnected system.

For the example of power system whose dynamics is represented by (7.5), the goal of low-level (self-feedback) control u_{s_i} is to ensure (input–output) stability of the

individual generator. The mid-level control u_{l_i} can be designed as a distributed optimization algorithm (by taking advantage of the communication network) to achieve a uniform voltage profile for a group of generators or minimize power loss. Finally, the high-level control v_i acts as a wide-area control with the goal of ensuring stability and/or improving performance of the power system.

In what follows, we will present a control theoretic framework based on dissipativity theory and cooperative control for systematically optimizing and controlling cyber-physical-human systems and further demonstrate its effectiveness using the power system example described previously.

7.3 Main Analytical and Design Tools: Dissipativity Theory and Cooperative Control

Dissipativity is an energy-like concept which describes input–output properties (e.g., stability) of a dynamical system. Input–output mapping becomes a useful way of quantifying input–output properties of the system when the dynamical model of the system is not available. Briefly speaking, dissipative system is a system that absorbs more energy from the external world than it supplies [23]. Passivity is a special class of dissipativity and is originated in circuit analysis. Passive systems are always decreasing in energy with respect to input energy. For example, an electrical circuit consisting of resistor, inductor, and capacitor can dissipate energy by turning it into heat and also store energy, but it cannot supply more energy than what has been put into it. Another class of dissipative systems is what so-called passivity-short systems. Compared to passive systems, passivity-short systems may increase or remain the same in energy from input to output during transience. One example is a generator that is not decreasing in energy at all times simply because it is producing some amount of energy. Dissipativity-based approaches become attractive in analyzing and controlling CPS since its properties are preserved over system interconnections which makes the approach computationally scalable. For example, with individual output negative feedback, the passivity-short systems can be interconnected either in parallel or in series or in a positive feedback loop or a negative feedback loop while maintaining the same passivity-short property [21]. This compositional property makes dissipativity a powerful and promising tool to analyze and control large-scale system such as CPS [2].

The concept of dissipativity is captured by introducing two energy-like functions, namely, supply rate and storage functions. Depending on the choice of particular supply rate function, dissipativity can imply several important behaviors such as stability of dynamical systems and their interconnections. Consider system (7.1) with $r_i = 0$ and without physical interconnection. The i-th system with supply rate $\Phi_i(u_i(t), y_i(t))$ is said to be dissipative if there exists a nonnegative real storage function $V_i(x_i)$ such that the following inequality holds [45]:

$$V_i(x_i(t)) - V_i(x_i(0)) \leqslant \int_0^t \Phi_i(u_i(\tau), y_i(\tau))d\tau. \tag{7.8}$$

Choosing the supply rate function in a quadratic form, the i-th system is said to be input passivity-short with respect to a differentiable storage function $V_i(x_i)$ if the inequality

$$\dot{V}_i \leqslant u_i^T y_i + \frac{\varepsilon_{ii}}{2}\|u_i\|^2 - \frac{\rho_i}{2}\|y_i\|^2 \tag{7.9}$$

holds for some $\varepsilon_{ii} > 0$, $\rho_i \geqslant 0$, and it is said to be output passivity-short if (7.9) holds for some $\varepsilon_{ii} \leqslant 0$, $\rho < 0$. In addition, the system is said to be L_2 stable if inequality (7.9) holds for some $\rho_i > 0$ and a positive definite V_i resulting in

$$\|y_i\|_{L_2} \leqslant \left(\frac{2\varepsilon_{ii}}{\rho_i} + \frac{4}{\rho_i^2}\right)\|u_i\|_{L_2} + \text{constant}. \tag{7.10}$$

Finally, the system is passive if inequality (7.9) holds for some $\varepsilon_{ii} = 0$ (and $\rho_i = 0$). Figure 7.3 illustrates a static input–output mapping of passivity and passivity-short systems. Note that passivity is quite restricted as it excludes most of linear dynamic systems such as nonminimum-phase systems and minimum-phase systems with relative degree 2 or higher. It is shown in [27] that most linear systems are passivity-short and that all linear Lyapunov-stable dynamic systems are either passivity-short or can be made passivity-short under an output-feedback control. The parameters ε_{ii} and ρ_i are important for analysis, control design, and stability of networked passivity-short systems, and it is desirable to maximize the value of ρ_i and minimize ε_{ii}. In particular, ε_{ii} is also called *impact coefficient* and it quantifies the impact of individual passivity-short system on the network-level cooperative control as will be discussed later. Let us show now that a synchronous generator connected to infinite bus is passivity-short. Dynamics of the generator is given by the following swing equation:

$$M_i\ddot{\delta}_i = b_i u_i - H_{ii}(\delta_i - \delta_i^*) - D_i\omega_0\dot{\delta}_i \tag{7.11}$$

and its output is defined as $y_i \triangleq \delta_i - \delta_i^*$. Taking the following positive definite storage function:
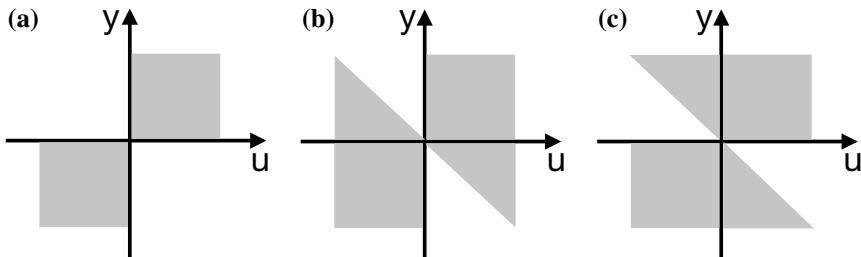
Fig. 7.3 Input–output diagram (shaded region) [22] of: **a** passive; **b** input passivity-short; **c** output passivity-short

$$V_i = \left(\frac{k_d}{2k\sqrt{kp}} + \frac{\sqrt{k_p}}{kk_d} \right) y_i^2 + \frac{1}{kk_d\sqrt{k_p}} \dot{y}_i^2 + \frac{1}{k\sqrt{k_p}} y_i \dot{y}_i$$

with $k = b_i/M_i$, $k_p = H_{ii}/M_i$, and $k_d = D_i\omega_0/M_i$ and computing its derivative yields

$$\dot{V}_i \leqslant u_i^T y_i + k \left(\frac{(1-\sqrt{k_p})^2}{2k_p\sqrt{k_p}} + \frac{1}{k_d^2\sqrt{k_p}} \right) u_i^2 - \frac{\sqrt{k_p}}{2k} y_i^2 \triangleq u_i^T y_i + \frac{\varepsilon_i}{2} \|u_i\|^2 - \frac{\rho_i}{2} \|y_i\|^2$$

which shows that the generator is passivity-short and L_2 stable. Furthermore, we can also obtain the physical meanings of ε_i and ρ_i. To this end, the transfer function of (7.11) can be written as

$$G(s) = \frac{k}{s^2 + k_d s + k_p}. \tag{7.12}$$

By writing $k_d = 2\xi\omega_n$, $k_p = \omega_n^2$, and $k \approx k_p$ where ω_n is the natural frequency and ξ denotes the damping ratio, it can be shown that

$$\varepsilon_i \approx \omega_n \left(1 - \frac{1}{\omega_n} \right)^2 + \frac{1}{2\xi^2\omega_n}, \quad \rho_i \approx \frac{1}{\omega_n}.$$

Hence, we can see that the value of ε_i increases as ξ becomes smaller and the optimal value of ε_i is obtained for $\omega_n = 1$.

Cooperative control is another control design tool that has shown a great promise in optimizing and controlling large-scale system and has been successfully utilized to develop network-level control of a group of mobile robots [3, 18], power system [54], charging scheduling of electric vehicles [16], and complex network [15]. The goal of cooperative control is to achieve nontrivial consensus using only local information (and thus scalable) obtained via the communication network as illustrated in Fig. 7.1, that is for all individual systems i, we have [42]

$$\lim_{t\to\infty} \|y_i(t) - y_j(t)\| = 0, \ \text{or} \ \lim_{t\to\infty} y_i(t) = c. \tag{7.13}$$

Consider again physically decoupled CPS with individual dynamics (7.1). As shown in [44], the concept of passivity-short simplifies the design of cooperative control by modularizing the lower level and network-level control designs. Specifically, a self-feedback control u_{s_i} is first designed so that individual system becomes passivity-short. The cooperative control can then be designed by simply considering the following fictitious integrator dynamics:

$$\dot{y}_i = u_{l_i} \tag{7.14}$$

where u_{l_i} is specified as

$$u_{l_i} = k_{y_i} \sum_{j \in \mathcal{N}_i^c} S_{ij}^c (y_j - y_i). \tag{7.15}$$

The closed-loop dynamics of (7.14) and (7.15) can be compactly written as

$$\dot{y} = -diag\{k_{y_1}, \ldots, k_{y_n}\} L y, \tag{7.16}$$

with $y = [y_1, \ldots, y_n]^T$ and $L = diag\{S^c \mathbb{1}\} - S^c$. Consensus (7.13) is ensured if there is at least one node from which every other node can be reached and the gains $k_{y_i} > 0$ are chosen to be smaller than k^*. Moreover, if every node can be reached from any other nodes, k^* can then be computed according to [44]

$$k^* = \frac{\lambda_2(\Gamma L + L^T \Gamma)}{2(\max_i \varepsilon_{ii})\lambda_{\max}(L^T \Gamma L)}, \tag{7.17}$$

where $\lambda_2(\cdot)$, $\lambda_{\max}(\cdot)$ denote the smallest nonzero and largest eigenvalues, respectively, and matrix $\Gamma = diag\{\eta_1\}$ with $\eta_1^T L = 0$. It is worth to note that k^* in (7.17) can be computed in a distributed manner without requiring global information of L [12]. The communication topology embedded in matrix L can also be optimized to increase the convergence speed of (7.15), see, e.g., [9, 11, 43]. As can be seen from (7.15) and (7.17), the design of cooperative control of networked passivity-short system does not require any explicit knowledge about the heterogeneous physical systems other than their impact coefficients. Moreover, quantity $\max_i \varepsilon_{ii}$ in (7.17) can be viewed as the "worst" value of impact coefficients of all the passivity-short systems. Adding or removing subsystems into or from the networked systems results in different impacts on the overall system operation. However, the performance of the overall system can still be guaranteed given that the control gains are appropriately upper bounded to limit such impact. Hence, the operation of the networked system can be performed in a plug-and-play manner while its stability is guaranteed.

7.4 Hierarchical Control Design for Cyber-Physical-Human Systems

In this section, we utilize the concept of passivity-short and cooperative control presented in the previous section to design hierarchical control law (7.7) for power system whose dynamics is given by (7.5).

7.4.1 Low-Level Control Design: Ensuring Input–Output Stability

Let us now consider the nominal subsystem in (7.5) by excluding its physical interconnections, i.e., assuming $H_{ik} = 0$ for all $i \neq k$. The first step is to design a self-feedback control u_{s_i} for individual physical system given by

$$u_{s_i} = -K_i x_i$$

such that: (i) the individual physical system is passivity-short and L_2 stable for input–output pair $\{\overline{u}_i, y_i\}$; (ii) its impact on the overall system, that is, the values ε_{ii} and $-\rho_i$ in (7.9) are minimized. To this end, taking the storage function $V = \frac{1}{2}x_i^T P_i x_i$ with P_i is a positive definite matrix, a self-feedback control can be designed by solving the following optimization problem:

$$
\begin{aligned}
\underset{K_i, \varepsilon_{ii}, \rho_i}{\text{minimize}} \quad & [\alpha_{ii}\varepsilon_{ii} - (1 - \alpha_{ii}\rho_i)] \\
\text{subject to} \quad & P_i > 0, \\
& M_i(x_i) \leqslant 0, \\
& \varepsilon_{ii}, \rho_i \geqslant 0,
\end{aligned}
\tag{7.18}
$$

where $\alpha_{ii} \in (0, 1)$ is a design parameter and matrix $M_i(x_i)$ is defined as

$$M_i(x_i) \triangleq (A_i(x_i) - B_i K_i)^T P_i + P_i(A_i(x_i) - B_i K_i) + \rho_i C_i^T C_i + \frac{1}{\varepsilon_{ii}}\|P_i B_i - C_i^T\|^2 < 0.$$

The second constraint in (7.18) guarantees that inequality (7.9) holds, i.e., the individual system is passivity-short and L_2 stable. Note that at any instant of time t, the state $x_i(t)$ becomes known from the Phasor Measurement Units (PMU) and so is matrix $A_i(x_i)$, and hence K_i can be designed adaptively by using available Lyapunov function $P_i > 0$.

After making the individual system passivity-short and L_2 stable, next we consider the interconnected system to quantify the impact of nonlinear interconnections on subsystem (7.5) in a way parallel to that of $\varepsilon_{ii}\|\overline{u}_i\|^2$. Specifically, the goal is to minimize the transient impacts of the inter-area oscillations encoded in ε_{ij} by solving the following optimization problem:

$$
\begin{aligned}
\underset{\varepsilon_{ij}}{\text{minimize}} \quad & \sum_{j \in \mathcal{N}_i} \alpha_{ij}\varepsilon_{ij} \\
\text{subject to} \quad & P_i > 0, \\
& M_i'(x_i, y_j) \leqslant 0, \\
& \varepsilon_{ij}, \alpha_{ij} \geqslant 0, \\
& \sum_{j \in \mathcal{N}_i} \alpha_{ij} = 1,
\end{aligned}
\tag{7.19}
$$

where

$$M_i' \triangleq M_i - \sum_{j \in \mathcal{N}_i} \left(P_i H_{ij} C_i + C_i^T H_{ij}^T P_i - \frac{1}{\varepsilon_{ij} P_i H_{ij} H_{ij}^T P_i} \right).$$

The second constraint in (7.19) guarantees that the following property holds:

$$\dot{V}_i \leqslant \overline{u_i}^T y_i + \frac{\varepsilon_{ii}}{2} \|\overline{u_i}\|^2 - \frac{\rho_i}{2} \|y_i\|^2 + \frac{1}{2} \sum_{j \in \mathcal{N}_i} \varepsilon_{ij} \|y_j\|^2,$$

where the terms $\varepsilon_{ij} \|y_j\|^2$ quantify the impact of nonlinear interconnections on the subsystem. Standard techniques to solve Linear or Bilinear Matrix Inequality [51] can be readily used to compute the solutions to both optimization problems (7.18) and (7.19).

7.4.2 Mid-level Control Design: Local Coordination Through Cyber-Physical Interconnection

Next, we design local coordination (cooperative) control u_{l_i} in (7.7) to improve the voltage profile of the power system. As a scenario, we consider a distribution network divided into several clusters as illustrated in Fig. 7.5. The goal is for the distributed generators (DGs) to cooperatively control their reactive power injection such that the sum of quadratic voltage errors of the DGs in each cluster is minimized. The problem can be formulated as the following optimization problem:

$$\min_{\vartheta_i} \sum_i f_i, \quad f_i = \frac{1}{2} (1 - E_i)^2, \tag{7.20}$$

where the control variable are DGs reactive power fair utilization ratios ϑ_i defined as $\vartheta_i = Q_{e_i} / \overline{Q}_{e_i}$ with $\overline{Q}_{e_i}$ denotes the maximum reactive power available to the i-th DG. The reactive power and voltage are coupled through the following power flow equation:

$$Q_{e_i} = -E_i^2 B_{ii} + \sum_{k \neq i} E_i E_k (G_{ik} \sin \delta_{ik} - B_{ik} \cos \delta_{ik}).$$

In addition, it is also desirable for the DGs in each cluster to contribute equally (i.e., the values ϑ_i reach a consensus for all DGs) in minimizing (7.20). To this end, the communication network is assumed to be bidirectional whose topology is similar to that of the distribution network as shown in Fig. 7.4. Cooperative control algorithm can then be designed to solve (7.20) as described in Section 7.3. Specifically, each DG adjusts its reactive power fair utilization ratio according to
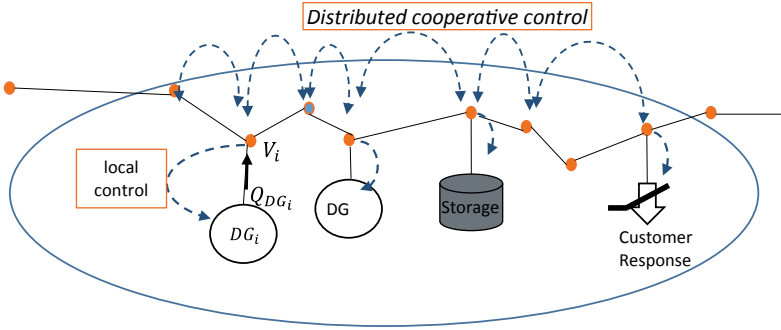
Fig. 7.4 Architecture of cooperative voltage control for distribution network as proposed in [33]
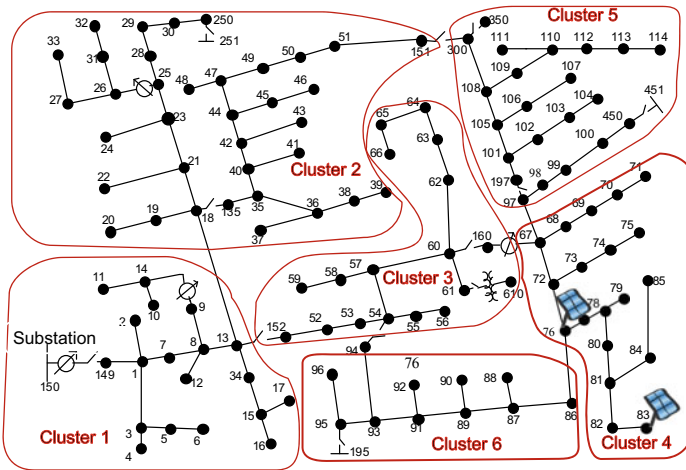
Fig. 7.5 A diagram of IEEE 123 bus system divided into six clusters

$$\dot{\vartheta}_i = u_{l_i} = \sum_{j \in \mathcal{N}_i^c} (\vartheta_j - \vartheta_i) - \beta_i \frac{\partial f_i}{\partial \vartheta_i}, \tag{7.21}$$

where $\beta_i > 0$ [33]. The first term of update rule (7.21) is a consensus protocol which facilitates the equal contribution of DGs into the reactive power generation while the second term corresponds to a (sub)gradient algorithm which minimizes the objective function in (7.20). Note that a similar strategy can also be applied to distributed frequency control with DGs as presented in [54].

We evaluate the performance of the cooperative control (7.21) using IEEE 123-bus test system divided into six clusters as shown in Fig. 7.5. The objective is to regulate the bus voltages in cluster 4 with two photovoltaics installed at buses 76 and 83, respectively. The voltage regulation using cooperative control (7.21) is compared with the one using droop control where the droop control gain is manually tuned

Fig. 7.6 Comparison of droop control and cooperative control strategies for regulating bus voltages in cluster 4
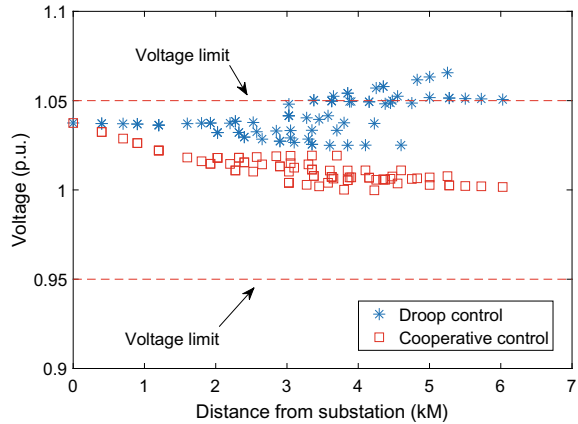
Fig. 7.7 Reactive power fair utilization ration for the DGs in cluster 4 under cooperative control (7.21)
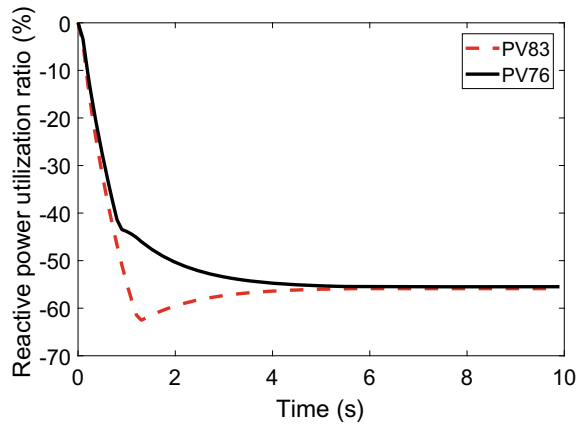
to achieve the best performance. Figure 7.6 shows the simulation results under both droop control and cooperative control strategies. As can be observed from the figure, droop control strategy results in voltage violations, that is, the voltage of the buses located far away from the substation exceeds the voltage limit of 1.05 p.u. On the other hand, using cooperative control (7.21), the voltage level can be successfully driven close to unity, and thus, the overvoltage problem can be eliminated. In addition, the cooperative control strategy also yields an equal reactive power fair utilization ratio for the DGs as shown in Fig. 7.7.

7.4.3 High-Level Control Design: Wide-Area Coordination

The final step is to design network-level control v_i in (7.7) to ensure the overall system stability and hence to effectively damp out potential inter-area oscillations.

As discussed in Section 7.3, the design of network-level control depends only on properties of individual subsystems, in particular their impact coefficient and L_2 parameter quantified by $\{\varepsilon_{ii}, \ldots, \varepsilon_{ij}, \ldots\}$ and ρ_i, respectively. Similar to (7.15), the wide-area control v_i is given by

$$v_i = k^w_{y_i} \sum_{j \in \mathcal{N}_i^w} S^w_{ij}(y_j - y_i), \qquad (7.22)$$

where matrix $S^w = [S^w_{ij}]$ represents the communication network of wide-area control. By choosing control gain $k^w_{y_i} \approx k_w$ and considering storage function $V^w = \sum_i \frac{\gamma_i}{k_w} V_i$, it can be shown by following similar steps as in [44] that system (7.5) exponentially converges to the desired output consensus provided that control gain k_w satisfies

$$-k_w L^T_w \Lambda L_w + (\Gamma_w L^T_w + L_w \Gamma_w) + \frac{\Phi}{k_w} \geqslant 0,$$

where

$$L_w = diag\{S^w \mathbb{1}\} - S^w, \; \Lambda = diag\{\varepsilon_{ii}\}, \; \Gamma_w = diag\{\gamma_i\}, \; \Phi = diag\{\phi_i\},$$

$$\phi_i = \gamma_i \rho_i - \sum_j \gamma_j \varepsilon_{ji}.$$

The proposed wide-area control is evaluated using a three-area power system as illustrated in Fig. 7.8. The simulation time is set to 60 s where at $t = 0.0$ s, a speed disturbance $\Delta = 0.01$ p.u. is added to the system. The wide-area control using cooperative control (7.22) is compared with the one using traditional control with typical design (constant gain). The simulation results of power angle for generator 3 for both control strategies are shown in Fig. 7.9. Even though the overall system is stable under both control strategies, it can be observed from the simulation results that by using the proposed cooperative control strategy, mitigation of the low-frequency oscillation (i.e., inter-area oscillation) is considerably improved in comparison to the oscillation under traditional control with constant gain. Note that similar results can also be observed for the other two generators in the power system.

7.5 Analysis of Human–Machine Interaction

Human interactions with the physical systems through the cyber components is a central aspect of cyber-physical-human systems. During the interactions, human may act as an operator such as in teleoperation [24] or semiautonomous robot control systems [3] in general. On the other hand, human may also perform as players or agents in multi-agent systems as can be observed in electricity market [39]. Therefore, it is important to formally and rigorously analyze the human–machine interactions (i.e.,
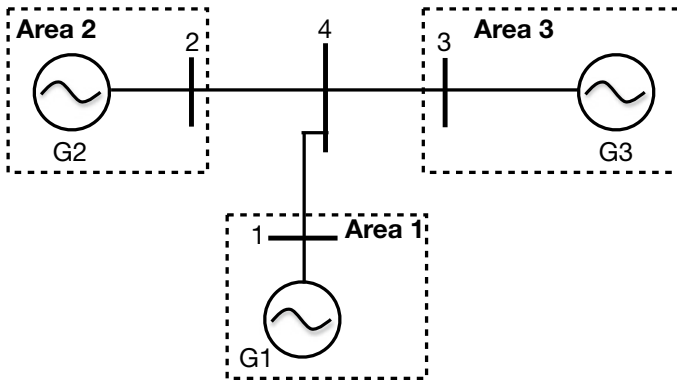
Fig. 7.8 A three-area power system
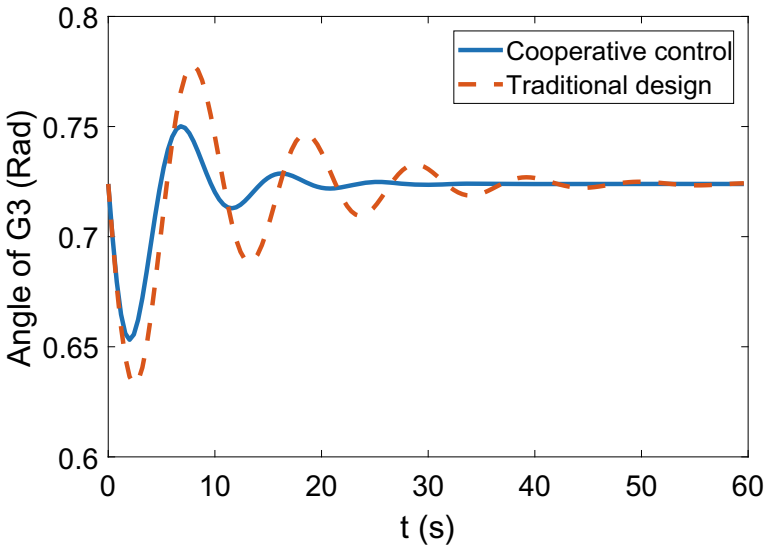
Fig. 7.9 Comparison of power angle for generator 3 under both cooperative wide-area control and traditional control strategies

human-in-the-loop control systems) in order to ensure the stability of the interconnected systems.

Dissipativity theory has been used to model the human decision-making and action in human–machine interactions due to its effectiveness in dealing with the largely unknown human dynamics and its modular design. For example, dissipativity-based modeling is developed and validated in [24] to model human arm endpoint characteristics in a human-teleoperated system. In addition, human–machine interactions in semiautonomous robotic swarm is modeled and analyzed in [3] using the concept

of passivity-short systems. In particular, it is theoretically shown and experimentally validated that human-operator modeled in [35] can be assumed to be a passivity-short system.

7.5.1 Human–Machine Interaction in Electricity Market

We focus on human as players or agents in multi-agent systems. As an example, we consider an electricity market consisting of multiple areas. In the i-th area, there are set of consumers, generators, and an independent system operator (ISO) engaged in electricity market trading. Specifically, the consumers and generators decide the amount of demand and power supply and the ISO uses the information to update the electricity price in each area as illustrated in Fig. 7.10. The goal is to maximize the profit of each market participant while balancing the supply and demand. The problem can be formulated as the following social welfare maximization problem:

$$
\begin{aligned}
\underset{P_L, P_G}{\text{maximize}} \quad & W(P_L, P_G) \\
\text{subject to} \quad & P_L = P_G, \\
& \text{other linear equality and inequality constraints,}
\end{aligned}
\tag{7.23}
$$

where W is the social welfare function which depends on the utility function (i.e., financial satisfaction) of both the consumers and generators, P_L, P_G are stacks of total electricity demand and supply in each area, respectively. Note that the solution to (7.23) may serve as the operational decision r_i in (7.1), see Fig. 7.2. The inequality constraints in (7.23) include upper and lower bounds on demand and supply. If the utility function of consumer and generator are strictly concave and convex functions, respectively, then optimization (7.23) has a unique solution. The convergence analysis of market trading to the solution of (7.23) can be viewed as stability analysis of the interconnected system of consumers, generators, and ISO as illustrated in Fig. 7.11.
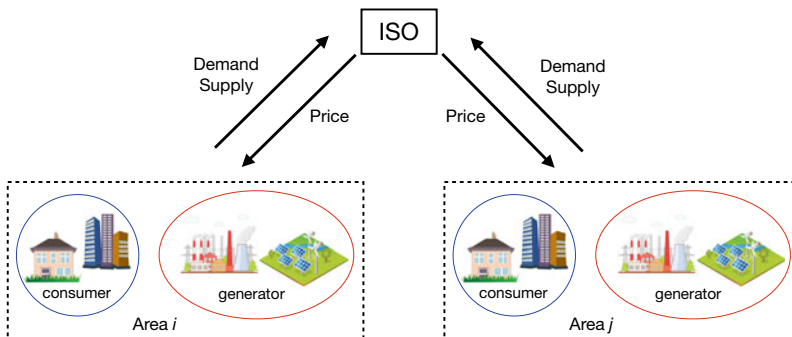
Fig. 7.10 Electricity market consisting of multiple areas
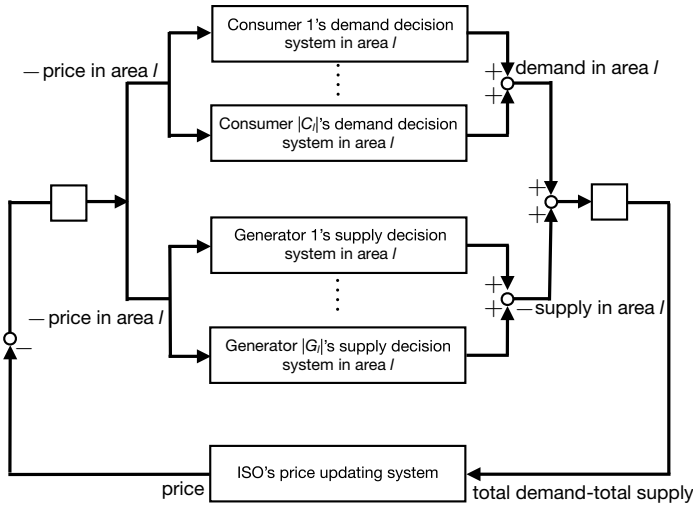
Fig. 7.11 Electricity market trading system in area l viewed as an interconnected system consisting of consumers, generators, and ISO
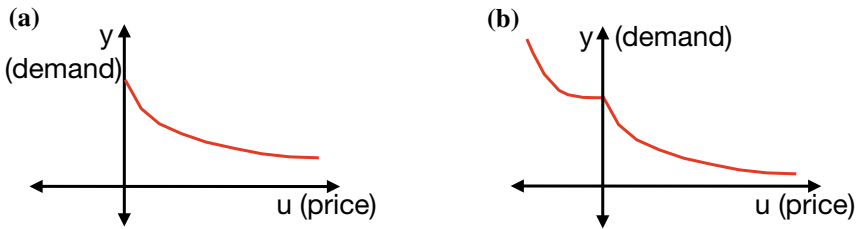
Fig. 7.12 Power demand curve with **a** normal (positive) price; **b** negative price

In particular, dynamics of consumer demand, generator supply decisions and ISO price updating in Fig. 7.11 can be obtained by applying dual decomposition to the dual problem of (7.23) where its Lagrange multiplier represents the (electricity) price [38]. When the power demand curve representing input–output static mapping between (positive) price and demand in electricity market is given in Fig. 7.12a, it is shown in [39] that each block's dynamics in Fig. 7.11 is (strictly) passive, and as a result, the interconnected system is also passive and hence stable. This means that the market trading system will converge to the optimal solution of (7.23).

However, the price in electricity market is not always positive especially when the number of renewable energy sources feeding into the power grid increases. For example, when high and inflexible power generation simultaneously appears and followed by low electricity demand, power prices may fall below zero (i.e., negative price) as can be often observed in Germany during public holidays such as Christmas. This means that power suppliers have to pay their customers to buy electric energy. The power demand curve when taking into account the negative price can be illustrated in Fig. 7.12b. Comparing the figure with input–output diagram in Fig. 7.3a,

it is obvious that dynamics of consumer demand decision system in Fig. 7.11 is not passive. It is shown in [38] that under power demand curve given in Fig. 7.12b, dynamics of consumer demand and generator supply decision systems in Fig. 7.11 are passivity-short as can be observed by comparing Figs. 7.3c and 7.12b. As a result, stability of the electricity market, i.e., interconnected system can still be guaranteed.

The discussions above focus on consumer demand decision dynamics derived from the (static) optimization problem (7.23). Another important issue is the analysis of human decision-making dynamics, that is, how the human responds (in terms of electricity demand) to the price change with main application to demand response (e.g., dynamic electricity pricing). There have been some efforts in dynamic modeling of price-responsive demand in electricity market using real data. For example, empirical study in [1] using data acquired at ERCOT suggests that (i) demand response during normal and peak price periods may have qualitatively different behavior, and (ii) there is a demand response delay on a high price surge. From the empirical study, we can initially observe that the dynamics of price-responsive demand is not a passive system due to the delay of the response. Further analysis is still required to investigate whether the dynamics exhibit passivity-short properties.

7.5.2   Transactive Control

The above example on electricity (competitive) market is a special case of *transactive control*. Transactive control is a new type of framework to coordinate a large number of distributed generations/loads by combining concepts from microeconomic theory and control theory [32]. Transactive control extends the concept of demand response to both the demand and supply sides whose objective is to balance via incentives (pricing) the supply and demand autonomously, in real-time and a decentralized manner [46]. In comparison to demand response such as price-responsive control and direct load control, transactive control preserves customer privacy and has more predictable and reliable aggregated load response. The potential of transactive control framework, in particular transactive energy system, has been demonstrated through several demonstration projects such as the Olympic Peninsula Demonstration [20] and AEP gridSMART demonstration [53]. Moreover, transactive control framework has been applied to manage distributed energy resources for different purposes such as congestion and voltage management [25, 26], providing spinning reserves [52], and residential energy management [37].

Broadly speaking, transactive control framework can be modeled using four key elements as proposed in [32]: payoff functions, control decisions, information, and solution concept. Consider a system consisting of $(n + 1)$ agents, that is one coordinator (agent 0) and n distributed energy resources (DERs) where each DER can communicate with each other and also with the coordinator to perform local decision-making. Local objective of both coordinator and DERs is represented by a payoff function U_i which depends on price μ_i and energy consumption p_i. Each DER aims at maximizing its own payoff function formulated as

$$\text{maximize} \quad U_i(\mu_i, p_i; \theta_i)$$
$$\text{subject to} \quad h_i(p_i; \theta_i) \leqslant 0,$$

where θ_i denotes private information of the agent such as preference and local constraints. Similarly, the coordinator aims to solve the following optimization problem:

$$\text{maximize} \quad U_0(\mu, p; \theta)$$
$$\text{subject to} \quad g(p, \mu; \theta) \leqslant 0; \quad h_i(p_i; \theta_i) \leqslant 0,$$

where $p = [p_1, \ldots, p_n]^T$, $\mu = [\mu_1, \ldots, \mu_n]^T$ and $\theta = [\theta_1, \ldots, \theta_n]^T$. Note that the payoff function of coordinator depends on prices and consumption of all DERs. Moreover, the coordinator also has a global constraint such as power flow constraint in the whole network. Next, to optimize the payoff functions, control decision are defined for each agent denoted by $\pi_i \in \Pi_i$ where Π_i is the feasible control decision of agent i. For example, by taking $\pi_0 = \mu$ and $\pi_i = p_i$ the payoff functions become $U_i(\pi_i, \pi_0; \theta_i)$ and $U_0(\pi_0, \pi_1, \ldots, \pi_n; \theta)$ which yields a coupling between decisions of DERs and coordinator. Another important element in transactive control is information set consisting of private information and information of control decision of each agent. Finally, information on control decisions provides a sequence of decision for the agents resulting in a multilevel decision problem. Within each layer, if the payoff function of each agent does not depend on decisions of other players then the solution is simply equal to the optimal solution to the standard optimization problem. On the other hand, if the payoff functions of each agent depends on the other agents, then we have a game problem whose solution corresponds to the game equilibrium. Two basic solution concepts to a game problem are Nash equilibrium (that is a collection of decisions from which no agent wants to deviate given that others stick to the equilibrium decision) and dominant strategy equilibrium (that is each agent will stick to the equilibrium strategy no matter what decisions other players make).

The four elements described above dictate the class of transactive problems (type of games) under consideration. For example, if the agent's payoff function is quasi-linear w.r.t. price and the coordinator's objective is to minimize the overall operational cost while satisfying some constraints, then we have a social maximization problem described in the previous subsection. On the other hand, if the payoff function is not quasi-linear and the coordinator's objective is different from maximizing the social welfare, we then have a Stackelberg game whose equilibrium computation is very challenging [6, 47, 49, 50].

Research challenges in transactive control include investigating price-response behavior of DERs and ensuring convergence of transaction control. For example, it is shown in [40] that a simple price strategy may stabilize the power system operation. Dissipativity theory provides a framework to systematically analyze this complex system as demonstrated in the previous subsection. Further research need to be performed to investigate the application of dissipativity theory for analyzing different transactive control problems.

7.6 Role of Real-Time Big Data and Decision-Making

The hierarchical control/optimization architecture presented in the previous subsections relies on real-time big data. Rapid development of sensor, wireless transmission, network communication technologies, smart devices, and cloud computing makes it possible to collect large amounts of data in real time. To illustrate further this point, let us take smart grid as an example. The main data source in smart grid is the advanced metering infrastructure (AMI) which deploys a large number of smart meters at the end-user side and collects, e.g., customers' electricity consumption data every 15 min [28, 56]. It is estimated that the amount of data collected by AMI will increase from 24 million a year to 220 million per day for a large utility company [56]. Moreover, the volume of data collected every 15 mins in a distribution network using 1 million devices will surge up to 2920 Tb [31]. In addition to AMI, PMUs are able to produce direct time-stamped voltage/current magnitudes and phase angle with sampling rate 30–60 samples per second, which is much faster than the data collection in Supervisory Control and Data Acquisition (SCADA) system [7]. As an illustration, the amount of data per day generated by100 PMUs with 20 measurements and at the sampling rate of 60 Hz is equal to 100 GB [30]. Other sources of big data in smart grid include weather data, mobile data, thermal sensing data, energy database, electric vehicle data, transmission line sensor, and dynamic pricing [56].

The increase of uncertainty (e.g., due to the high renewable energy penetration) and tight interconnection between and within the layers calls for real-time processing and decision-making. To this end, big data can be utilized for developing novel real-time learning, optimization, and decision-making (control) algorithms for cyber-physical-human systems as illustrated in the previous sections. For example, big data has many applications in the operation of smart grid [48]. A new algorithms using PMU data is proposed in [8] to accelerate the state estimation process. Moreover, a PMU based robust estimation method is presented in [55] to eliminate unwanted perturbed data and thus increases the robustness of state estimation algorithm. Big data can also be used for fault detection and classification in micro-grid leading to a much better performance compared to model-based approach [36]. AMI and other sensors provide opportunity to realize line impedance calibration (i.e., parameters) for distribution power system which was not possible previously [41]. Weather data can also be used for predicting the power generation of renewable energy sources such as wind turbines which further can be utilized for voltage control and demand response [19]. Furthermore, with the exponentially increasing number of PMUs deployed, and the resulting explosion in data volume, wide-area measurement systems (WAMS) technology as the key to guaranteeing stability, reliability, situational awareness, state estimation, and control of next-generation power systems is bound to transcend from centralized to a distributed architecture within the next few years. Motivated by this fact, a distributed optimization based learning algorithm is proposed in [10] for one of the most critical wide-area monitoring applications—namely, estimation of mode shapes for inter-area oscillation modes.

The exposure to external network such as Internet comes at a price of data security and privacy [29, 34]. Cyber incidents or network intrusion may cause physical damage to the physical system due to the tight coupling between the physical system and the cyber-layer. Unfortunately, traditional security solutions in the ICT (information and communications technology) domain are not sufficient to ensure security and resilience of the network since they do not take into account the physical attacks through direct interaction with the components in physical systems. For example, by placing a shunt around a meter the integrity of a meter can be violated without the need of breaking the cybersecurity countermeasure. They may also introduce adverse effects on the operation of CPS. For example, while cryptography can enhance the confidentiality of data flows, it may result in unacceptable time latency and degrade the performance of time-critical functionalities in CPS. Moreover, coordinated network attacks by sophisticated adversaries undermine standard residual based detection schemes. It is discussed in [13, 14] that control theoretic framework together with recent advancement in cloud computing and network management (e.g., software defined networking) show promises in ensuring the resilient operation of CPS against (coordinated and intelligent) cyberattacks.

7.7 Conclusion

The chapter presents a scalable and modular control-theoretical framework to model, analyze, optimize, and control cyber-physical-human systems. It is shown that efficient computational algorithms can be applied hierarchically to operate and optimize cyber-physical-human systems, first individually to quantify the dynamic behavior of every agent, then locally to describe the local interactions of neighboring agents, and finally to the overall system. All the three control levels deal with real-time big data, and the hierarchical structure makes the overall optimization and control problem scalable and solvable. In particular, we present and highlight two main tools whose combination shows a great promise to optimize and control such tightly interconnected system. The first tool is the concept of dissipativity theory which is a useful way of quantifying input–output properties of dynamical systems and whose compositional property makes it a powerful tool to analyze and control CPS. The second tool is cooperative control which allows the designer to develop a scalable and robust optimization and control algorithms. Application to power system is investigated as an illustrative example.

References

1. An, J., Kumar, P., Xie, L.: On transfer function modeling of price responsive demand: an empirical study. In: IEEE Power and Energy Society General Meeting, pp. 1–5 (2015)
2. Antsaklis, P.J., Goodwine, B., Gupta, V., McCourt, M.J., Wang, Y., Wu, P., Xia, M., Yu, H., Zhu, F.: Control of cyberphysical systems using passivity and dissipativity based methods. Eur. J. Control. **19**(5), 379–388 (2013)
3. Atman, M.W.S., Hatanaka, T., Qu, Z., Chopra, N., Yamauchi, J., Fujita, M.: Motion synchronization for semi-autonomous robotic swarm with a passivity-short human operator. Int. J. Intell. Robot. Appl. **2**(2), 235–251 (2018)
4. Baheti, R., Gill, H.: Cyber-physical systems. Impact Control Technol. **12**(1), 161–166 (2011)
5. Cassandras, C.G.: Smart cities as cyber-physical social systems. Engineering **2**(2), 156–158 (2016)
6. Colson, B., Marcotte, P., Savard, G.: An overview of bilevel optimization. Ann. Oper. Res. **153**(1), 235–256 (2007)
7. DOE: Advancement of synchrophasor technology in ARRA projects. https://www.smartgrid.gov/recovery_act/program_publications.html
8. Göl, M., Abur, A.: A fast decoupled state estimator for systems measured by PMUs. IEEE Trans. Power Syst. **30**(5), 2766–2771 (2015)
9. Gusrialdi, A.: Performance-oriented communication topology design for distributed control of interconnected systems. Math. Control Signals Syst. **25**(4), 559–585 (2013)
10. Gusrialdi, A., Chakrabortty, A., Qu, Z.: Distributed learning of mode shapes in power system models. In: IEEE Conference on Decision and Control, pp. 4002–4007 (2018)
11. Gusrialdi, A., Qu, Z.: Growing connected networks under privacy constraint: achieving trade-off between performance and security. In: IEEE Conference on Decision and Control, pp. 312–317 (2015)
12. Gusrialdi, A., Qu, Z.: Distributed estimation of all the eigenvalues and eigenvectors of matrices associated with strongly connected digraphs. IEEE Control Syst. Lett. **1**(2), 328–333 (2017)
13. Gusrialdi, A., Qu, Z.: Smart grid security: attacks and defenses. In: Stoustrup, J., Annaswamy, A., Chakrabortty, A., Qu, Z. (eds.) Smart Grid Control: An Overview and Research Opportunities, pp. 199–223. Springer (2018)
14. Gusrialdi, A., Qu, Z.: Towards resilient operation of smart grid. In: Stoustrup, J., Annaswamy, A., Chakrabortty, A., Qu, Z. (eds.) Smart Grid Control: An Overview and Research Opportunities, pp. 275–288. Springer (2018)
15. Gusrialdi, A., Qu, Z., Hirche, S.: Distributed link removal using local estimation of network topology. IEEE Trans. Netw. Sci. Eng. (2018)
16. Gusrialdi, A., Qu, Z., Simaan, M.A.: Distributed scheduling and cooperative control for charging of electric vehicles at highway service stations. IEEE Trans. Intell. Transp. Syst. **18**(10), 2713–2727 (2017)
17. Gusrialdi, A., Qu, Z., Simaan, M.A.: Competitive interaction design of cooperative systems against attacks. IEEE Trans. Autom. Control **63**(9), 3159–3166 (2018)
18. Gusrialdi, A., Yu, C.: Exploiting the use of information to improve coverage performance of robotic sensor networks. IET Control Theory Appl. **8**(13), 1270–1283 (2014)
19. Haghi, H.V., Qu, Z.: A Kernel-based predictive model of EV capacity for distributed voltage control and demand response. IEEE Trans. Smart Grid **9**(4), 3180–3190 (2018)
20. Hammerstrom, D.J., Ambrosio, R., Carlon, T.A., DeSteese, J.G., Horst, G.R., Kajfasz, R., Kiesling, L.L., Michie, P., Pratt, R.G., Yao, M., et al.: Pacific northwest gridwise? Testbed demonstration projects; Part I. Olympic Peninsula Project. Technical report, Pacific Northwest National Lab. (PNNL), Richland, WA (United States) (2008)
21. Harvey, R., Qu, Z.: Cooperative control and networked operation of passivity-short systems. In: Control of Complex Systems, pp. 499–518. Elsevier (2016)
22. Harvey, R., Xu, Y., Qu, Z., Namerikawa, T.: Dissipativity-based design of local and wide-area DER controls for large-scale power systems with high penetration of renewables. In: IEEE Conference on Control Technology and Applications, pp. 2180–2187 (2017)

23. Hill, D., Moylan, P.: The stability of nonlinear dissipative systems. IEEE Trans. Autom. Control **21**(5), 708–711 (1976)
24. Hirche, S., Buss, M.: Human-oriented control for haptic teleoperation. Proc. IEEE **100**(3), 623–647 (2012)
25. Hu, J., You, S., Lind, M., Ostergaard, J.: Coordinated charging of electric vehicles for congestion prevention in the distribution grid. IEEE Trans. Smart Grid **5**(2), 703–711 (2014)
26. Ipakchi, A.: Demand side and distributed resource management—a transactive solution. In: IEEE Power and Energy Society General Meeting, pp. 1–8 (2011)
27. Joo, Y., Harvey, R., Qu, Z.: Cooperative control of heterogeneous multi-agent systems in a sampled-data setting. In: IEEE Conference on Decision and Control, pp. 2683–2688 (2016)
28. Karnouskos, S., Terzidis, O., Karnouskos, P.: An advanced metering infrastructure for future energy networks. In: New Technologies, Mobility and Security, pp. 597–606. Springer (2007)
29. Khurana, H., Hadley, M., Lu, N., Frincke, D.A.: Smart-grid security issues. IEEE Secur. Priv. **8**(1) (2010)
30. Klump, R., Agarwal, P., Tate, J.E., Khurana, H.: Lossless compression of synchronized phasor measurements. In: IEEE Power and Energy Society General Meeting, pp. 1–7 (2010)
31. Lavastorm: Big data, analytics, and energy consumption. http://www.lavastorm.com/blog/2012/04/09/big-data-analytics-and-energy-consumption/
32. Lian, J., Zhang, W., Sun, Y., Marinovici, L.D., Kalsi, K., Widergren, S.E.: Transactive system: Part I: Theoretical underpinnings of payoff functions, control decisions, information privacy, and solution concepts. Technical report, Pacific Northwest National Lab. (PNNL), Richland, WA (United States) (2018)
33. Maknouninejad, A., Qu, Z.: Realizing unified microgrid voltage profile and loss minimization: a cooperative distributed optimization and control approach. IEEE Trans. Smart Grid **5**(4), 1621–1630 (2014)
34. McDaniel, P., McLaughlin, S.: Security and privacy challenges in the smart grid. IEEE Secur. Priv. **3**, 75–77 (2009)
35. McRuer, D.: Human dynamics in man-machine systems. Automatica **16**(3), 237–253 (1980)
36. Mishra, D.P., Samantaray, S.R., Joos, G.: A combined wavelet and data-mining based intelligent protection scheme for microgrid. IEEE Trans. Smart Grid **7**(5), 2295–2304 (2016)
37. Moradzadeh, B., Tomsovic, K.: Two-stage residential energy management considering network operational constraints. IEEE Trans. Smart Grid **4**(4), 2339–2346 (2013)
38. Muto, K., Namerikawa, T., Qu, Z.: Passivity-short-based stability analysis on electricity market trading system considering negative price. In: IEEE Conference on Control Technology and Applications, pp. 418–423 (2018)
39. Okawa, Y., Namerikawa, T., Qu, Z.: Passivity-based stability analysis of dynamic electricity pricing with power flow. In: IEEE Conference on Decision and Control, pp. 813–818 (2017)
40. Pentland, A.: Economics: simple market models fail the test. Nature **525**(7568), 190 (2015)
41. Peppanen, J., Reno, M.J., Broderick, R.J., Grijalva, S.: Distribution system model calibration with big data from AMI and PV inverters. IEEE Trans. Smart Grid **7**(5), 2497–2506 (2016)
42. Qu, Z.: Cooperative Control of Dynamical Systems. Springer, London (2009)
43. Qu, Z., Simaan, M.: An analytic solution to the optimal design of information structure and cooperative control in networked systems. In: IEEE Conference on Decision and Control, pp. 4015–4022 (2012)
44. Qu, Z., Simaan, M.A.: Modularized design for cooperative control and plug-and-play operation of networked heterogeneous systems. Automatica **50**(9), 2405–2414 (2014)
45. Sepulchre, R., Jankovic, M., Kokotovic, P.: Constructive Nonlinear Control. Springer, London (1997)
46. Sijie, C., Chen-Ching, L.: From demand response to transactive energy: state of the art. J. Mod. Power Syst. Clean Energy **5**(1), 10–19 (2017)
47. Soliman, H.M., Leon-Garcia, A.: Game-theoretic demand-side management with storage devices for the future smart grid. IEEE Trans. Smart Grid **5**(3), 1475–1485 (2014)
48. Tu, C., He, X., Shuai, Z., Jiang, F.: Big data issues in smart grid—a review. Renew. Sustain. Energy Rev. **79**, 1099–1107 (2017)

49. Tushar, W., Saad, W., Poor, H.V., Smith, D.B.: Economics of electric vehicle charging: a game theoretic approach. IEEE Trans. Smart Grid **3**(4), 1767–1778 (2012)
50. Tushar, W., Zhang, J.A., Smith, D.B., Poor, H.V., Thiébaux, S.: Prioritizing consumers in smart grid: a game theoretic approach. IEEE Trans. Smart Grid **5**(3), 1429–1438 (2014)
51. VanAntwerp, J.G., Braatz, R.D.: A tutorial on linear and bilinear matrix inequalities. J. Process Control **10**(4), 363–385 (2000)
52. Weckx, S., D'hulst, R., Driesen, J.: Primary and secondary frequency support by a multi-agent demand control system. IEEE Trans. Power Syst. **30**(3), 1394–1404 (2015)
53. Widergren, S.E., Subbarao, K., Fuller, J.C., Chassin, D.P., Somani, A., Marinovici, M.C., Hammerstrom, J.L.: AEP Ohio gridSMART demonstration project real-time pricing demonstration analysis. PNNL Rep. **23192** (2014)
54. Xin, H., Qu, Z., Seuss, J., Maknouninejad, A.: A self-organizing strategy for power flow control of photovoltaic generators in a distribution network. IEEE Trans. Power Syst. **26**(3), 1462–1473 (2011)
55. Zhao, J., Zhang, G., Das, K., Korres, G.N., Manousakis, N.M., Sinha, A.K., He, Z.: Power system real-time monitoring by using PMU-based robust state estimation method. IEEE Trans. Smart Grid **7**(1), 300–309 (2016)
56. Zhou, K., Fu, C., Yang, S.: Big data driven smart energy management: from big data to big insights. Renew. Sustain. Energy Rev. **56**, 215–225 (2016)

Chapter 8
Distributed Optimization Based Control on the Example of Microgrids

Philipp Braun, Philipp Sauerteig and Karl Worthmann

Abstract Model Predictive Control (MPC) is nowadays one of the most successful advanced process control methodologies and is used in a wide range of applications. While originally limited to processes with slow dynamics and a limited number of states, the applicability of MPC schemes increased dramatically over the past years due to the performance of modern microchips and the concurrent advancements of mathematical optimization, in particular, distributed optimization. In this paper, we outline the ideas of distributed optimization schemes embedded in MPC implementations on the example of the dual ascent algorithm and the alternating direction method of multipliers. The performance and the properties of the resulting distributed optimization based control schemes are illustrated on the example of a network of distributed energy systems. In particular, the overall power demand of the network is optimized by using flexibilities resulting from distributed storage devices and controllable loads.

8.1 Introduction

Model Predictive Control (MPC) is nowadays the most successful advanced process control methodology, which can be concluded from its wide range of applications [1, 2]. The key factors for its success are the simplicity of the basic idea—measure/estimate the current state, predict and optimize the future plant behavior to compute an input signal, and repeat this procedure ad infinitum—and its capability to deal with constrained nonlinear multi-input multi-output systems. For details on

P. Braun (✉)
School of Electrical Engineering and Computing, University of Newcastle, Newcastle, Australia
e-mail: philipp.braun@newcastle.edu.au

P. Sauerteig · K. Worthmann
Institute for Mathematics, Technische Universität Ilmenau, Ilmenau, Germany
e-mail: philipp.sauerteig@tu-ilmenau.de

K. Worthmann
e-mail: karl.worthmann@tu-ilmenau.de

MPC as well as the development of different MPC concepts over the past decades, we refer to the monographs [3, 4], the review article [5], and the references therein.

While the range of applications was limited due to the real-time requirements on the optimization step in MPC, see, e.g., [6], recent developments [7, 8] in mathematical programming allowed to further extend the scope of application. Herein, distributed optimization, see [9–11], plays a major role to overcome the computational limitations. To this end, new distributed MPC schemes have been developed [12, 13] to embed iterative distributed optimization algorithms within the MPC loop such that performance guarantees can be concluded [14–16]. The potential of distributed optimization in prediction based control is outlined in the review article [17] and in the book [18].

The interplay of MPC and iterative distributed optimization schemes is both the starting point and the focus of our work. After a brief introduction, we give a conceptual review of dual decomposition to explicate the main ideas behind distributed optimization. The section is concluded by presenting and discussing a dual ascent algorithm [19], specified in terms of an optimal control problem for distributed dynamical systems. While dual ascent already allows for a significant speed-up, see, e.g., [14], and is very flexible with respect to the structure of the interconnection of dynamical systems, it requires—in general—restrictive assumptions for convergence to the (global) optimum. This drawback can, e.g., be mitigated by using the Alternating Direction Method of Multipliers (ADMM), which is concisely outlined in the successor section. We refer to [11] for a recently published more detailed exposition. The key point demonstrated by doing so is that distributed optimization algorithms are typically tailored for a particular setting, in which their performance—in combination with other features like plug-and-play capability—is extremely competitive with respect to scalability, which ensures that the algorithm remains computationally tractable also for large-scale systems.

In this paper, the proposed algorithms are tailored to a smart-grid application to illustrate their properties within the MPC closed loop. In the context of the application, the overall energy demand of a network of Residential Energy Systems (RESs) is optimized. In this field of application, distributed optimization is nowadays very popular, see, e.g., [20, 21] or the review article [22] for further comments on the application. We consider a setting in which RESs are connected to the grid via a Central Entity (CE), an operator of a transmission grid, for example. Since the number of RESs is typically large, distributed optimization is used to alleviate the computational burden resulting from solving the optimization problems in MPC, and to maintain flexibility with respect to changes in the network structure of the smart grid. Here, we stick to the term Optimal Control Problem (OCP) despite the fact that we consider discrete-time systems, which implies that the OCP is a finite dimensional optimization problem. This work continues our first modeling approach (accompanied by first numerical results) on the integration of controllable loads [23]. Using ADMM, we rigorously show that the proposed distributed optimization approach converges to the global minimum and, thus, lay the foundation for its embedding in distributed MPC. From an application point of view exploiting flexibilities resulting from both energy storage devices *and* controllable loads further improves the contribution to

load shaving necessary due to the integration of renewable (and, thus, highly fluctuating) energy sources. Moreover, we investigate the distributed MPC scheme applied to optimize the *energy exchange* between microgrids coupled through transmission lines. In particular, we numerically show that an exchange, on top of the optimization within the individual microgrids, is beneficial with respect to peak shaving even if losses due to the transmission are taken into account. Again, distributed optimization (ADMM) turns out to be the right tool to leverage the untapped potential resulting from this additional *flexibility*.

Throughout the paper, $\mathbb{N}_0$ and $\mathbb{N}$ denote the natural numbers including and excluding 0, and $\mathbb{Z}$ and $\mathbb{R}$ denote the integers and the real numbers, respectively. For given $n, m \in \mathbb{Z}$, $n \leq m$, we use $[n : m] = \{n, n+1, \ldots, m\}$ to define the integers from n to m to shorten the expressions. The identity matrix of appropriate dimension is denoted by I. For a sequence $\{a_1, a_2, a_3, \ldots\} \subset \mathbb{R}^n$ the shorter notation $(a^l)_{l \in \mathbb{N}} \subset \mathbb{R}^n$ is used. For a vector $x \in \mathbb{R}^n$, $n \in \mathbb{N}$, $\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$ denotes the Euclidean norm.

8.2 Model Predictive Control

To illustrate the MPC concept, we consider a discrete time control system governed by the dynamics

$$x(k + 1) = f(x(k), u(k)), \qquad x(0) = x^0, \tag{8.1}$$

with a continuous map $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$. Here, $x(k)$ and $u(k)$ denote the state of the system and the control input at time instant k, $k \in \mathbb{N}_0$, respectively. In addition, we assume that the states and the control have to respect certain constraints described through compact sets with nonempty interior, i.e., $x(k) \in \mathbb{X} \subset \mathbb{R}^{n_x}$ and $u(k) \in \mathbb{U} \subset \mathbb{R}^{n_u}$, $k \in \mathbb{N}_0$.

With these definitions, MPC is used to determine a state feedback law $\mu : \mathbb{N}_0 \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}$, through stage costs $\ell : \mathbb{N}_0 \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ defining a performance measure, such that the closed-loop system

$$x^{\text{cl}}(k + 1) = f(x^{\text{cl}}(k), \mu(k, x^{\text{cl}}(k))), \qquad x^{\text{cl}}(0) = x^0 \tag{8.2}$$

is *recursively feasible* and satisfies *infinite horizon performance* properties.

- *Recursive feasibility* refers to properties of the closed-loop solution $x^{\text{cl}}(k)$, $k \in \mathbb{N}_0$. If $x^{\text{cl}}(k)$ is feasible, i.e., $x^{\text{cl}}(k) \in \mathbb{X}$ holds, the current feedback value and the successor state are also feasible, i.e.,

$$\mu(k, x^{\text{cl}}(k)) \in \mathbb{U} \qquad \text{and} \qquad x^{\text{cl}}(k + 1) \in \mathbb{X}. \tag{8.3}$$

Thus, recursive feasibility implies well-posedness of the closed loop (8.2) provided initial feasibility $x^{\text{cl}}(0) = x^0 \in \mathbb{X}$ holds, see, e.g. [4, 24].

- The *infinite horizon performance* sums up the stage cost along the MPC closed-loop trajectory

$$\sum_{k=0}^{\infty} \ell(k, x^{\text{cl}}(k), \mu(k, x^{\text{cl}}(k))) \tag{8.4}$$

in cases where the limit exists. If the limit does not exist, or in applications where the average is more meaningful, the infinite horizon performance can alternatively be measured by

$$\limsup_{T \to \infty} \frac{1}{T} \sum_{k=0}^{T-1} \ell(k, x^{\text{cl}}(k), \mu(k, x^{\text{cl}}(k))). \tag{8.5}$$

The performance index (8.4) is typically used in set point stabilization, see, e.g. [3, 4], while the average (8.5) is often used in *economic MPC*, see, e.g. [25, 26]. The main difference is that the stage cost is positive definite with respect to a steady state for set point stabilization while in economic MPC turnpike properties [27] in combination with (strict) dissipativity [28] are used to deduce rigorous assertions with respect to the closed-loop performance.

In general, the problem of finding a feedback law that is optimal regarding the infinite horizon performances (8.4) or (8.5) is computationally intractable. MPC is a technique to approximately solve this task. To this end, an optimization problem on a predefined finite time horizon $N \in \mathbb{N}_{\geq 2}$ is considered in a receding horizon fashion, leading to the following basic MPC scheme. Here, the term *receding horizon* means that the feedback law μ is computed iteratively at each time step $k \in \mathbb{N}_0$ (and only for the current state $x(k)$).

Basic MPC Scheme

1. For $k \in \mathbb{N}_0$, measure the current state $\hat{x} := x(k)$.
2. Minimize the cost functional

$$J_N(k, \hat{x}, \mathbf{u}) := \sum_{n=k}^{k+N-1} \ell(n, x(n), u(n)) \tag{8.6}$$

where $\mathbf{u} = (u(k), u(k+1), \dots, u(k+N-1))^T$ and subject to the initial condition $x(k) = \hat{x}$, the system dynamics

$$x(n+1) = f(x(n), u(n)), \quad n \in [k : k+N-1], \tag{8.7}$$

and the state and control constraints

$$x(n+1) \in \mathbb{X}, \quad u(n) \in \mathbb{U}, \quad n \in [k : k+N-1], \tag{8.8}$$

to compute a minimizing sequence $\mathbf{u}^\star$.

3. Implement $\mu(k, \hat{x}) := u^\star(k)$, shift the horizon forward in time, i.e., set $k = k + 1$, and go to Step 1.

While the basic MPC algorithm is well understood, already time varying stage costs ℓ lead to a significantly more involved analysis, see, e.g. [29, 30]. Moreover, a careful design of the stage cost ℓ is important to rigorously deduce closed-loop properties as shown in [31, 32].

In this paper, we want to apply the idea of the MPC scheme to a distributed setting. To be more precise, let $\mathscr{I} \in \mathbb{N}_{\geq 2}$ dynamically decoupled systems

$$x_i(k + 1) = f_i(x_i(k), u_i(k)), \qquad x_i(0) = x_i^0, \qquad i \in [1 : \mathscr{I}], \qquad (8.9)$$

be given where $x_i \in \mathbb{X}_i \subset \mathbb{R}^{n_{x_i}}$, $u_i \in \mathbb{U}_i \subset \mathbb{R}^{n_{u_i}}$, and the dimensions of the overall system are given by $n_x = \sum_{i=1}^{\mathscr{I}} n_{x_i}$, and $n_u = \sum_{i=1}^{\mathscr{I}} n_{u_i}$, respectively.

Even though the individual systems are not coupled through their dynamics, we assume that the performance of the overall dynamics depends on the individual decisions taken by the subsystems $i \in [1 : \mathscr{I}]$. To incorporate this into the model, we assume the presence of a Central Entity (CE) such that each subsystem may communicate with the CE (star-shaped topology) via

$$z_i(k) = h_i(k, x_i(k), u_i(k)) \qquad (8.10)$$

with communication variables $z_i \in \mathbb{R}^p$ for all $i \in [1 : \mathscr{I}]$. The functions $h_i : \mathbb{N}_0 \times \mathbb{R}^{n_i} \times \mathbb{R}^{m_i} \to \mathbb{R}^p$ are assumed to be continuous and can depend on the current time step $k \in \mathbb{N}_0$ as well as the local states and the local control, for all $i \in [1 : \mathscr{I}]$. With these definitions the coupling between the individual systems can be described through global stage costs

$$\hat{\ell} : \mathbb{N}_0 \times \mathbb{R}^{p\mathscr{I}} \to \mathbb{R}, \qquad (k, z) \mapsto \hat{\ell}(k, z) \qquad (8.11a)$$

involving the communication variables $z := (z_1, \ldots, z_{\mathscr{I}})^T$. The stage costs (8.11a) capture the performance of the overall system, or equivalently, they capture the costs enforced by the CE. In addition, every system $i \in [1 : \mathscr{I}]$ can have local performance measures defined through local stage costs

$$\ell_i : \mathbb{N}_0 \times \mathbb{R}^{n_i} \times \mathbb{R}^{m_i} \to \mathbb{R}, \qquad (k, x_i, u_i) \mapsto \ell_i(k, x_i, u_i) \quad \forall i \in [1 : \mathscr{I}], \quad (8.11b)$$

which in combination with the global stage costs (8.11a) lead to multi-objective optimization problems [33] or multi-objective MPC formulations [34] in the control context.

The overall Optimal Control Problem (OCP) to be solved at every time instant $k \in \mathbb{N}_0$ in Step 2 of the MPC scheme can be summarized by the following optimization problem where the notation $\mathbf{x} := (x(k), x(k+1), \ldots, x(k+N-1))^T$, $\mathbf{u} := (u(k), u(k+1), \ldots, u(k+N-1))^T$ is used.

Minimize the cost functional

$$J_N(k, \hat{x}, \mathbf{u}) := \sum_{n=k}^{k+N-1} \left(\alpha \cdot \hat{\ell}(n, z(n)) + \frac{1-\alpha}{\mathscr{I}} \cdot \sum_{i=1}^{\mathscr{I}} \ell_i(n, x_i(n), u_i(n)) \right) \quad (8.12)$$

w.r.t. $u(k), u(k+1), \ldots, u(k+N-1)$ subject to $x(k) = \hat{x}$,

the equalities (8.9) and (8.10), the state and control constraints

$x_i(n+1) \in \mathbb{X}_i(n), u_i(n) \in \mathbb{U}_i(n)$ for $n \in \{k, k+1, \ldots, k+N-1\}$

The parameter $\alpha \in [0, 1]$, which we assume to be set and defined by the CE, shifts the emphasis on the performance of the individual subsystems encoded in ℓ_i, $i \in [1 : \mathscr{I}]$, to a network-wide objective $\hat{\ell}$ by choosing α between 0 and 1. The overall distributed MPC algorithm incorporating the OCP (8.12) is summarized in Algorithm 1.

Algorithm 1 Distributed MPC algorithm

Input: Time horizon $N \in \mathbb{N}$, number of systems $\mathscr{I} \in \mathbb{N}$, stage costs $\hat{\ell}$, ℓ_i, $i \in [1 : \mathscr{I}]$, and a weighting parameter $\alpha \in [0, 1]$.
Initialization: Set $k = 0$.
Main loop: For $k \in \mathbb{N}_0$

(1) For all $i \in [1 : \mathscr{I}]$, measure the current states $\hat{x}_i := x_i(k)$.
(2) Solve the OCP (8.12) to obtain minimizing sequences

$$\mathbf{u}_i^\star = (u_i^\star(k), \ldots, u_i^\star(k+N-1))^T \quad \forall i \in [1 : \mathscr{I}]. \quad (8.13)$$

(3) For all $i \in [1 : \mathscr{I}]$, implement $\mu_i(k, \hat{x}_i) := u_i^\star(k)$, shift the horizon forward in time, i.e., set $k = k + 1$, and go to Step (1).

While Algorithm 1 can be easily written down, Step (2) requires particular care due to the real-time requirements in MPC and to maintain the decoupled structure of the individual subsystems. Hence, the goal of the next section is to outline algorithms, which can be suitably adapted such that Step (2) is computationally tractable and flexible with respect to changes in local system dynamics. Here, iterative distributed optimization schemes play a major role to mitigate the computational burden on the one hand and, on the other hand, allow for a premature stop after a few iterations while still ensuring feasibility and achieving a competitive performance.

8.3   Distributed Optimization

To compute an MPC feedback law, the OCP (8.12) of the form

$$\min_{u_i(n), x_i(n), z_i(n)} \sum_{n=k}^{k+N-1} \left(\alpha \cdot \hat{\ell}(n, (z_1, \ldots, z_{\mathscr{I}})^T(n)) + \tfrac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \ell_i(n, x_i(n), u_i(n)) \right) \tag{8.14a}$$

$$\text{s.t.}\quad x_i(k) = \hat{x}_i, \tag{8.14b}$$

$$x_i(n+1) = f_i(x_i(n), u_i(n)), \tag{8.14c}$$

$$z_i(n) = h_i(n, x_i(n), u_i(n)), \tag{8.14d}$$

$$x_i(n+1) \in \mathbb{X}_i(n+1), \quad u_i(n) \in \mathbb{U}_i(n), \tag{8.14e}$$

$$\forall\, i \in [1 : \mathscr{I}], \quad \forall\, n \in [k : k+N-1].$$

has to be solved at every time instant k, $k \in \mathbb{N}_0$. To this end, an efficient optimization algorithm is essential to meet the real-time requirements of the MPC scheme proposed in Algorithm 1. For applications with a large number of systems $\mathscr{I}$, $\mathscr{I} \in \mathbb{N}$, scalability and flexibility with respect to $\mathscr{I}$ are indispensable properties for the algorithm design, see also [35, 36]. The latter also includes changes in the particular system dynamics of the individual systems. In this context, iterative distributed optimization schemes regained considerable attention over the last years.

In this section, we discuss distributed optimization schemes, and in particular hierarchical distributed optimization schemes, and their embedding within the MPC Algorithm 1. In Section 8.3.1, we discuss the basic dual decomposition algorithm. While the convergence results are restrictive, dual decomposition nicely illustrates the main idea of distributed optimization in general. The Alternating Direction Method of Multipliers (ADMM) presented in Section 8.3.2 extends the ideas of dual decomposition and is less restrictive with respect to the stage costs ℓ. We conclude this chapter in Section 8.3.3 by discussing beneficial properties of iterative optimization schemes within MPC.

Since we are interested in the solution of a static optimization problem (8.14a–8.14e) in this section, we simplify the notion before we continue. We collect the information of the variables x_i, u_i, z_i, $i \in [1 : \mathscr{I}]$, over the prediction horizon at time $k \in \mathbb{N}$ in a single variable

$$\mathbf{y} = \begin{pmatrix} \mathbf{y_1} \\ \vdots \\ \mathbf{y}_{\mathscr{I}} \end{pmatrix} \quad \text{with} \quad \mathbf{y}_i = \begin{pmatrix} x_i(k) & u_i(k) & z_i(k) \\ \vdots & \vdots & \vdots \\ x_i(k+N-1) & u_i(k+N-1) & z_i(k+N-1) \end{pmatrix}.$$

Even though the variables $z_i(n)$, $i \in [1 : \mathscr{I}]$, $n \in [k : k+N-1]$, are contained in the definition of the variable $\mathbf{y}$ we additionally define

$$\mathbf{z} = \begin{pmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_{\mathscr{I}} \end{pmatrix} \in \mathbb{R}^{pN\mathscr{I}} \quad \text{with} \quad \mathbf{z}_i = \begin{pmatrix} z_i(k) \\ \vdots \\ z_i(k+N-1) \end{pmatrix}$$

to collect the data of the shared variables. For simplicity of exposition, the k-dependency is dropped in the variables $\mathbf{y}$ and $\mathbf{z}$ in this section.

Remark 8.1 Note that the variables $\mathbf{y}_i$ only need to contain the information of the variables x_i, u_i, z_i. By doing so, we do not restrict ourselves to a particular formulation of the optimization problem. For example in MPC problems, where the dynamics (8.1) are linear, it is common to optimize with respect to $\mathbf{u}$ and to remove the state vector $\mathbf{x}$ from the set of unknowns, see, e.g. [3, Chapter 12.1].

With these definitions of the vector of unknowns, the stage costs can be summarized in the functions

$$\phi_i(\mathbf{y}_i; k) = \sum_{n=k}^{k+N-1} \ell_i(n, x_i(n), u_i(n)) \quad \forall i \in [1 : \mathscr{I}] \tag{8.15}$$

$$\text{and} \quad \hat{\phi}(\mathbf{z}; k) = \sum_{n=k}^{k+N-1} \hat{\ell}(n, (z_1, \dots, z_{\mathscr{I}})^T(n)). \tag{8.16}$$

Similarly, the constraints (8.14b) to (8.14e) are summarized in the sets

$$\mathbb{D}_i(k) = \left\{ \mathbf{y}_i \left| \begin{array}{l} x_i(k) = \hat{x}_i, \\ x_i(n+1) = f_i(x_i(n), u_i(n)), \\ z_i(n) = h_i(n, x_i(n), u_i(n)), \\ x_i(n+1) \in \mathbb{X}_i(n+1), \quad u_i(n) \in \mathbb{U}_i(n), \\ \quad \forall n \in [k : k+N-1]. \end{array} \right. \right\} \tag{8.17}$$

for all $i \in [1 : \mathscr{I}]$ and $\mathbb{D}(k) = \mathbb{D}_1(k) \times \cdots \times \mathbb{D}_{\mathscr{I}}(k)$. Combining all these definitions, the OCP (8.14a) can simply be written as

$$\min_{\mathbf{y}_i \in \mathbb{D}_i} \quad \alpha \cdot \hat{\phi}(\mathbf{z}; k) + \frac{1-\alpha}{\mathscr{I}} \cdot \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i; k). \tag{8.18}$$

Coupling of the variables takes place in the function $\hat{\phi}$. Otherwise, the optimization problem (8.18) could be split up into $\mathscr{I}$ independent optimization problems. Next, we present a solution technique for the optimization problem based on iterative distributed optimization and explicit conditions on the function ϕ and the set $\mathbb{D}(k)$ such that convergence to the global solution is guaranteed.

8.3.1 Dual Decomposition

Dual decomposition is a relatively old concept with its roots in the 1960s and is discussed in many papers and monographs. Here, we follow the exposition given in the monographs [10, 37] to introduce concepts for our particular setting and refer to these books for details and references.

The coupling in the objective function (8.14a) can be eliminated by introducing a new variable $\mathbf{a} = \mathbf{z}$ (clearly also $\mathbf{a}_i = \mathbf{z}_i$ holds for all $i \in [1 : \mathscr{I}]$) linked to the Central Entity (CE) and, then, shifting the coupling from the objective function to the constraints

$$\min_{\mathbf{y}_i \in \mathbb{D}_i(k), \, \mathbf{a} \in \mathbb{R}^{\mathscr{I} pN}} \quad \alpha \hat{\phi}(\mathbf{a}; k) + \tfrac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i; k) \tag{8.19a}$$

$$\text{s.t.} \quad \mathbf{z}_i - \mathbf{a}_i = 0 \quad \forall \, i \in [1 : \mathscr{I}]. \tag{8.19b}$$

The advantage of the reformulation is that in the Lagrangian

$$\mathscr{L}(\mathbf{y}, \mathbf{a}, \lambda; k) = \alpha \hat{\phi}(\mathbf{a}; k) + \tfrac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i; k) + \sum_{i=1}^{\mathscr{I}} \lambda_i^T (\mathbf{z}_i - \mathbf{a}_i) \tag{8.20}$$

$$= \left(\alpha \hat{\phi}(\mathbf{a}; k) - \sum_{i=1}^{\mathscr{I}} \lambda_i^T \mathbf{a}_i \right) + \sum_{i=1}^{\mathscr{I}} \left(\tfrac{1-\alpha}{\mathscr{I}} \phi_i(\mathbf{y}_i; k) + \lambda_i^T \mathbf{z}_i \right)$$

the optimization variables $\mathbf{a}$ and $\mathbf{y}$ are separated in independent functions. Thus, under appropriate conditions on the objective functions and the local constraints, an optimal solution to the primal problem (8.19) can be obtained by solving the (unrestricted) dual problem

$$\max_{\lambda \in \mathbb{R}^{p\mathscr{I}N}} \psi(\lambda; k) \quad \text{with dual function } \psi(\lambda; k) = \inf_{\substack{\mathbf{y} \in \mathbb{D}(k) \\ \mathbf{a} \in \mathbb{R}^{\mathscr{I} pN}}} \mathscr{L}(\mathbf{y}, \mathbf{a}, \lambda; k). \tag{8.21}$$

The following theorem characterizes the interplay of the primal and the dual problem, see, e.g. [10, Appendix C].

Theorem 8.1 *If the primal problem (8.19) has an optimal solution* $(\mathbf{y}^\star, \mathbf{a}^\star)$*, then also the dual problem (8.21) has an optimal solution* $\lambda^\star$ *(and vice versa) and the optimal values coincide, i.e.,*

$$\alpha \hat{\phi}(\mathbf{a}^\star; k) + \tfrac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i^\star; k) = \psi(\lambda^\star; k). \tag{8.22}$$

Moreover, the pair $(\mathbf{y}^\star, \mathbf{a}^\star, \lambda^\star)$ *is optimal for (8.19) and (8.21) if and only if the saddle point condition*

$$\mathscr{L}(\mathbf{y}^\star, \mathbf{a}^\star, \lambda; k) \leq \mathscr{L}(\mathbf{y}^\star, \mathbf{a}^\star, \lambda^\star; k) \leq \mathscr{L}(\mathbf{y}, \mathbf{a}, \lambda^\star; k) \tag{8.23}$$

is satisfied for all $\mathbf{y} \in \mathbb{D}(k)$*,* $\mathbf{a} \in \mathbb{R}^{p\mathscr{I}N}$ *and for all* $\lambda \in \mathbb{R}^{p\mathscr{I}N}$*.*

As a consequence of Theorem 8.1, if an optimal solution to the dual problem (8.21) is known, an optimal solution of the primal problem can be computed by solving the optimization problems

$$\mathbf{y}_i^\star \in \underset{\mathbf{y}_i \in \mathbb{D}_i(k)}{\arg\min} \frac{1-\alpha}{\mathscr{I}} \phi_i(\mathbf{y}_i; k) + \mathbf{z}_i^T \lambda_i^\star \tag{8.24}$$

for all $i \in [1 : \mathscr{I}]$ in parallel. An optimal solution of the dual problem can be found by iteratively updating the Lagrange multipliers. Hence, under appropriate conditions on the primal problem (8.19) an optimal solution $(\mathbf{y}^\star, \mathbf{a}^\star)$ and $\lambda^\star$ can be computed through the hierarchical dual ascent Algorithm 2.

Algorithm 2 Hierarchical dual ascent algorithm

Input: Step size $c \in \mathbb{R}_0$, number of systems $\mathscr{I} \in \mathbb{N}$, objective functions $\hat{\phi}, \phi_i, i \in [1 : \mathscr{I}]$, weighting parameter $\alpha \in [0, 1]$, and maximal number of iterations $l_{max} \in \mathbb{N}$.
Initialization: Set $l = 0$ and initialize $\lambda^0 \in \mathbb{R}^{p\mathscr{I} N}$ (arbitrarily).
Main loop: For $l \leq l_{max}$

1. The individual systems compute an optimal solution of

$$\mathbf{y}_i^l \in \underset{\mathbf{y}_i \in \mathbb{D}_i(k)}{\arg\min} \frac{1-\alpha}{\mathscr{I}} \phi_i(\mathbf{y}_i; k) + \mathbf{z}_i^T \lambda_i^l \quad \forall i \in [1 : \mathscr{I}] \tag{8.25a}$$

   in parallel and send $\mathbf{z}_i^l$ to the CE.
2. The CE computes an optimal solution of

$$\mathbf{a}^l \in \underset{\mathbf{a} \in \mathbb{R}^{p\mathscr{I} N}}{\arg\min} \alpha\hat{\phi}(\mathbf{a}; k) - \sum_{i=1}^{\mathscr{I}} \mathbf{a}_i^T \lambda_i^l. \tag{8.25b}$$

3. The CE updates the Lagrange multipliers

$$\lambda_i^{l+1} = \lambda_i^l + c(\mathbf{z}_i^l - \mathbf{a}_i^l) \quad \forall i \in [1 : \mathscr{I}], \tag{8.25c}$$

   and broadcasts λ_i^{l+1}. Afterwards, set $l = l + 1$ and go to Step 1.

Convergence of Algorithm 2 can, e.g., be guaranteed by Theorem 8.2.

Theorem 8.2 ([37, Proposition 1.2.3]) *Consider the primal problem (8.19) and assume that the objective function is strongly convex[1] with respect to the variables* $(\mathbf{y}, \mathbf{a})$ *with parameter* $\chi > 0$. *Moreover, assume that the primal problem (8.19) is feasible, and the sets* $\mathbb{D}_i(k)$ *are convex and closed for all* $i \in [1 : \mathscr{I}]$. *Then, for a constant stepsize* $c \in (0, \chi)$ *the iterates* $(\mathbf{y}^l)_{l\in\mathbb{N}}, (\mathbf{a}^l)_{l\in\mathbb{N}}, (\lambda^l)_{l\in\mathbb{N}}$ *computed according to Algorithm (8.25) satisfy*

$$\lim_{l\to\infty} \mathbf{z}^l = \mathbf{z}^\star, \quad \lim_{l\to\infty} \mathbf{a}^l \to \mathbf{a}^\star, \quad \lim_{l\to\infty} \lambda^l \to \lambda^\star. \tag{8.26}$$

---

[1]A function $F : \mathbb{R}^n \to \mathbb{R}$ is said to be *strongly convex* with parameter $\chi > 0$ if $F(\mu x + (1 - \mu)y) \leq \mu F(x) + (1 - \mu)F(y) - \frac{\chi}{2}\mu(1 - \mu)\|x - y\|_2^2$ holds for all $x, y \in \mathbb{R}^n$ and all $\mu \in [0, 1]$.

for arbitrary initial values $\mathbf{y}^0 \in \mathbb{D}(k)$, $\mathbf{a}^0 \in \mathbb{R}^{p\mathscr{I}N}$, $\lambda^0 \in \mathbb{R}^{p\mathscr{I}N}$.

The name *dual ascent* refers to the fact that (8.25c) is a gradient step in the direction of the dual function ψ, where $\psi(\lambda^{l+1}) > \psi(\lambda^l)$ holds if the condition on the stepsize is satisfied. The maximal stepsize c depends on the convexity of the objective function as well as on the definition of the linear coupling constraints (8.19b), see [10, Proposition 6.1.1] for details.

The dual ascent Algorithm 2 allows to split the optimization problem in tasks performed by the individual systems in parallel (see Algorithm 2, Step 1), and tasks performed by a CE (Steps 2 and 3). The local optimization problems (8.25a) depend on local information and on the dual variables λ_i, $i \in [1 : \mathscr{I}]$. Thus, the dimension and the complexity of the optimization problem solely depends on the parameters defining the ith system, $i \in [1 : \mathscr{I}]$. Note that also the number of systems $\mathscr{I} \in \mathbb{N}$ does not need to be known by the individual systems. To achieve this, the scaled Lagrange multipliers $\widetilde{\lambda}_i = \lambda_i \mathscr{I}$, $i \in [1 : \mathscr{I}]$, are broadcast by the CE and individual systems need to solve the optimization problem

$$\mathbf{y}_i^l \in \underset{\mathbf{y}_i \in \mathbb{D}_i(k)}{\arg\min} \quad (1-\alpha)\phi_i(\mathbf{y}_i; k) + \mathbf{z}_i^T \widetilde{\lambda}_i^l \tag{8.27}$$

for $i \in [1 : \mathscr{I}]$, instead of (8.25a).

The dimension of the optimization problem (8.25b) linked to the CE depends on the number of systems $\mathscr{I} \in \mathbb{N}$. Nevertheless, note that the variables $\mathbf{a} \in \mathbb{R}^{p\mathscr{I}N}$ are unconstrained and thus an optimal solution can be computed efficiently or even explicitly by the CE.

Note that depending on the network structure and the interconnection of the individual systems, the dimension of the variable $\mathbf{a}$ and the dimension of the Lagrange multipliers λ can be reduced (see [10, 11], for example), or in special cases they can even be made independent of the number of systems $\mathscr{I} \in \mathbb{N}$. Consider for example stage costs $\hat{\ell}$, which can be written in the form

$$\hat{\ell}\left(n; (z_1, \ldots, z_{\mathscr{I}})^T\right) = \overline{\ell}\left(n; \tfrac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} z_i\right) \tag{8.28}$$

and thus, the stage costs only depend on the average taken over the communication variables z_i, $i \in [1 : \mathscr{I}]$. Hence, we can define the variables $\overline{a}(n) = \tfrac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} z_i(n)$ for $n \in [k : k + N - 1]$, which are the elements of the vector $\overline{\mathbf{a}}$ and modify the cost function (8.16):

$$\overline{\phi}(\overline{\mathbf{a}}; k) = \hat{\phi}(\mathbf{z}; k) = \sum_{n=k}^{k+N-1} \overline{\ell}\left(n; \tfrac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} z_i(n)\right). \tag{8.29}$$

Then, the primal optimization problem (8.19) becomes

$$\min_{\substack{\mathbf{y}_i \in \mathbb{D}_i(k) \\ \overline{\mathbf{a}} \in \mathbb{R}^{pN}}} \alpha\overline{\phi}(\overline{\mathbf{a}}; k) + \tfrac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i; k) \quad \text{s.t.} \quad \tfrac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \mathbf{z}_i - \overline{\mathbf{a}} = 0 \tag{8.30}$$

and the dual ascent scheme (8.25) is of the form

$$\mathbf{y}_i^l \in \underset{\mathbf{y}_i \in \mathbb{D}_i(k)}{\arg\min}\ (1 - \alpha)\phi_i(\mathbf{y}_i; k) + \mathbf{z}_i^T \overline{\lambda}^l \qquad \forall\, i \in [1 : \mathscr{I}], \tag{8.31a}$$

$$\overline{\mathbf{a}}^l \in \underset{\overline{\mathbf{a}} \in \mathbb{R}^{pN}}{\arg\min}\ \alpha \overline{\phi}(\overline{\mathbf{a}}; k) - \sum_{i=1}^{\mathscr{I}} \overline{\mathbf{a}}_i^T \overline{\lambda}^l, \tag{8.31b}$$

$$\overline{\lambda}^{l+1} = \overline{\lambda}^l + c\left(\tfrac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \mathbf{z}_i^l - \overline{\mathbf{a}}^l\right). \tag{8.31c}$$

Here, the dimension of $\overline{\mathbf{a}} \in \mathbb{R}^{pN}$ as well as the dimension of the Lagrange multipliers $\overline{\lambda} \in \mathbb{R}^{p\mathscr{I}}$ is independent of the the number of systems. Moreover, the same information is communicated from the CE to the individual systems. In this case, the stepsize c needs to be chosen such that $c \in (0, 2\mathscr{I}\chi/[2 + \mathscr{I}])$ is satisfied to guarantee convergence [38].

Overall, the dual ascent Algorithm 2 thus provides a very flexible scheme to solve OCPs embedded in the MPC Algorithm 1. However, the assumed strong convexity of the objective function is very restrictive.

8.3.2   The Alternating Direction Method of Multipliers

To weaken the assumption on strong convexity in Theorem 8.2, and thus to extend the applicability of iterative distributed optimization schemes, alternative algorithms based on the augmented Lagrangian $\mathscr{L}_\rho(\mathbf{y}, \mathbf{a}, \lambda; k)$ defined by

$$\begin{aligned}\mathscr{L}_\rho(\mathbf{y}, \mathbf{a}, \lambda; k) = \alpha\hat{\phi}(\mathbf{a}; k) + \tfrac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i; k) \\ + \sum_{i=1}^{\mathscr{I}} \left(\lambda_i^T(\mathbf{z}_i - \mathbf{a}_i) + \tfrac{\rho}{2}\|\mathbf{z}_i - \mathbf{a}_i\|_2^2\right),\end{aligned} \tag{8.32}$$

for a positive parameter $\rho > 0$, have been proposed. For $\rho = 0$ the original definition of the Lagrangian (8.20) is recovered. Observe that for a feasible solution of the optimization problem (8.19) the quadratic terms vanish. In contrast to the Lagrangian, the variables $\mathbf{z}$ and $\mathbf{a}$ are not decoupled (or separable) in the augmented Lagrangian. However, it is still possible, to optimize the variables $\mathbf{z}$ and $\mathbf{a}$ sequentially, which leads to the ADMM Algorithm 3 and in particular to the iteration scheme (8.33).

The ADMM scheme received a lot of attention over the last years, especially due to the exposition and discussion of the algorithm in [11] and [39]. The advantage of ADMM compared to dual ascent is that convergence can be shown under weaker assumptions and independently of the stepsize.

Algorithm 3 Alternating Direction Method of Multipliers (ADMM)

Input: Step size $\rho \in \mathbb{R}_{>0}$, number of systems $\mathscr{I} \in \mathbb{N}$, objective functions $\hat{\phi}$, ϕ_i, $i \in [1 : \mathscr{I}]$, a weighting parameter $\alpha \in [0, 1]$, and a maximal number of iterations $l_{\max} \in \mathbb{N}$.
Initialization: Set $l = 0$ and initialize λ^0, $\mathbf{a}^0 \in \mathbb{R}^{p\mathscr{I}N}$ (arbitrarily).
Main loop: For $l \leq l_{\max}$

1. The individual systems compute an optimal solution of

$$\mathbf{y}_i^{l+1} \in \underset{\mathbf{y}_i \in \mathbb{D}_i(k)}{\arg\min} \frac{1-\alpha}{\mathscr{I}} \phi_i(\mathbf{y}_i; k) + \mathbf{z}_i^T \lambda_i^l + \frac{\rho}{2} \left\| \mathbf{z}_i - \mathbf{a}_i^l \right\|_2^2 \qquad \forall i \in [1 : \mathscr{I}] \qquad (8.33a)$$

 in parallel and broadcast $\mathbf{z}_i^{l+1}$ to the CE.
2. The CE computes an optimal solution of

$$\mathbf{a}^{l+1} \in \underset{\mathbf{a} \in \mathbb{R}^{p\mathscr{I}N}}{\arg\min} \alpha\hat{\phi}(\mathbf{a}; k) + \sum_{i=1}^{\mathscr{I}} -\mathbf{a}_i^T \lambda_i^l + \frac{\rho}{2} \left\| \mathbf{z}_i^{l+1} - \mathbf{a}_i \right\|_2^2. \qquad (8.33b)$$

3. The CE updates the Lagrange multipliers

$$\lambda_i^{l+1} = \lambda_i^l + \rho(\mathbf{z}_i^{l+1} - \mathbf{a}_i^{l+1}) \qquad \forall i \in [1 : \mathscr{I}] \qquad (8.33c)$$

 and broadcasts $(\lambda_i^{l+1}, \mathbf{a}_i^{l+1})$ to system $i \in [1 : \mathscr{I}]$. Afterwards, set $l = l + 1$ and go to Step 1.

Theorem 8.3 ([11, Section 3.2.1]) *Let the functions $\hat{\phi}$, ϕ_i, $i = [1 : \mathscr{I}]$, be convex. Suppose there exists a saddle point $(\mathbf{y}^\star, \mathbf{a}^\star, \lambda^\star)$ of the unaugmented Lagrangian $\mathscr{L}_0$, i.e., $(\mathbf{y}^\star, \mathbf{a}^\star, \lambda^\star)$ satisfies (8.23) for all $\mathbf{y} \in \mathbb{D}(k)$, $\mathbf{a}, \lambda \in \mathbb{R}^{p\mathscr{I}N}$. Then, for $\mathbf{y}^0 \in \mathbb{D}(k)$, $\mathbf{a}^0 \in \mathbb{R}^{p\mathscr{I}N}$ and $\lambda^0 \in \mathbb{R}^{p\mathscr{I}N}$ and fixed stepsizes $\rho > 0$, Algorithm 3 satisfies the following properties:*

1. *The sequence $(\mathbf{z}^l - \mathbf{a}^l)_{l \in \mathbb{N}}$ converges to zero, ensuring feasibility of the optimization problem (8.19).*
2. *The sequence $\left(\alpha\hat{\phi}(\mathbf{a}^l; k) + \frac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i^l; k) \right)_{l \in \mathbb{N}}$ converges to the optimal value of the optimization problem (8.19).*
3. *The dual variables $(\lambda^l)_{l \in \mathbb{N}}$ converge to the optimal dual point $\lambda^\star$.*

Theorem 8.3 weakens the assumptions on the objective function of Theorem 8.2 from strong convexity to convexity. The saddle point condition in Theorem 8.3 is, e.g., satisfied if the sets $\mathbb{D}_i(k)$, $i \in [1 : \mathscr{I}]$, are convex and compact. Compared to the dual ascent Algorithm 2, Steps 1 and 2 of Algorithm 2 need to be performed sequentially and cannot be performed in parallel.

Remark 8.2 In Algorithm 3, also the primal variables $\mathbf{a} \in \mathbb{R}^{p\mathscr{I}N}$ need to be communicated. However, the information contained in λ and $\mathbf{a}$ can be compressed in a single communication variable of dimension $p\mathscr{I}N$, which allows to achieve the same communication costs as in Algorithm 2, see the scaled version of ADMM proposed in [11, Section 3.1.1].

Even though it is not as obvious as in the dual ascent Algorithm 2, also the ADMM scheme allows a formulation of Algorithm 3 where the dimension of $\mathbf{a}$ and

λ is independent of the number of systems $\mathscr{I} \in \mathbb{N}$ if an objective function of the form (8.29) is used. In this case, the iteration scheme (8.33) becomes

$$\mathbf{y}_i^{l+1} \in \underset{\mathbf{y}_i \in \mathbb{D}_i(k)}{\arg\min} \ \frac{1-\alpha}{\mathscr{I}} \phi_i(\mathbf{y}_i; k) + \frac{\rho}{2} \left\| \mathbf{z}_i - \mathbf{z}_i^l + \Pi^l \right\|_2^2 \qquad \forall i \in [1 : \mathscr{I}] \tag{8.34a}$$

$$\overline{\mathbf{a}}^{l+1} \in \underset{\overline{\mathbf{a}} \in \mathbb{R}^{pN}}{\arg\min} \ \alpha \overline{\phi}(\overline{\mathbf{a}}; k) + \frac{\rho \mathscr{I}}{2} \left\| \frac{\overline{\lambda}^l}{\rho} + \frac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \mathbf{z}_i^{l+1} - \overline{\mathbf{a}} \right\|_2^2 \tag{8.34b}$$

$$\overline{\lambda}^{l+1} = \overline{\lambda}^l + \rho \left(\frac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \mathbf{z}_i^{l+1} - \overline{\mathbf{a}}^{l+1} \right) \tag{8.34c}$$

$$\Pi^{l+1} = \frac{\overline{\lambda}^{l+1}}{\rho} + \left(\frac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \mathbf{z}_i^{l+1} - \overline{\mathbf{a}}^{l+1} \right). \tag{8.34d}$$

See [11, Section 7] or [40] for a derivation of the iteration scheme, for example.

8.3.3 Properties Within the MPC Closed Loop

Algorithms 2 and 3 enable us to solve the OCPs (8.12) by iteratively solving smaller or less complex optimization problems independent of the number of systems. Thus, the numerical complexity of the distributed optimization algorithms strongly depends on the number of iterations needed to compute a solution within a predefined tolerance. If a good *initial guess*, e.g. initial values λ^0 and $\mathbf{a}^0$ close to the optimal solution, are available, the number of iterations is typically much smaller. Here, MPC provides a natural way to initialize λ^0 and $\mathbf{a}^0$ since at two consecutive time instants k and $k+1$, $k \in \mathbb{N}_0$, similar optimization problems are solved and only the first piece of the optimal trajectory is used to define a feedback law, while the rest is discarded. To illustrate this fact let the last iteration of the distributed optimization algorithm at time k be denoted by $\lambda_i^{l_{\max}}[k]$ and the initial value at time $k+1$ by $\lambda_i^0[k+1]$ for $i \in [1 : \mathscr{I}]$. Then, we get

$$\lambda_i^{l_{\max}}[k] = \begin{pmatrix} \lambda_i^{l_{\max}}(k|k) \\ \lambda_i^{l_{\max}}(k+1|k) \\ \vdots \\ \lambda_i^{l_{\max}}(k+N-2|k) \\ \lambda_i^{l_{\max}}(k+N-1|k) \end{pmatrix} \quad \rightsquigarrow \quad \begin{pmatrix} \lambda_i^{l_{\max}}(k+1|k) \\ \vdots \\ \lambda_i^{l_{\max}}(k+N-2|k) \\ \lambda_i^{l_{\max}}(k+N-1|k) \\ 0 \end{pmatrix} = \lambda_i^0[k+1].$$

The variable $\mathbf{a}$ can be initialized in the same way. Observe that in Algorithm 2 the variables $\mathbf{a}$ and $\mathbf{y}$ do not need to be initialized since Steps 1 and 2 in the first iteration $l = 0$ only depend on the initialization of λ^0. The same holds for the variable $\mathbf{y}$ in the ADMM algorithm. Typically, already a few iterations are enough to obtain a closed-loop performance, which is close to optimal.

Moreover, note that even though the iterates $\mathbf{z}_i^l, i \in [1 : \mathscr{I}], l \in [0 : l_{\max}]$ might not be optimal, feasibility (i.e., $\mathbf{z}_i \in \mathbb{D}_i(k), i \in [1 : \mathscr{I}]$) is assured in every iteration since

by assumption the systems are physically decoupled. Thus, early termination of the distributed optimization algorithm always provides a feasible solution if $\mathbb{D}(k) \neq \emptyset$ holds.

8.4 Case Study: Distributed MPC for Small Scale Electricity Networks

In this section, we apply the distributed MPC Algorithm 1, where the optimization Step (2) is carried out with the ADMM Algorithm 3 to a small scale electricity network. The input variables (controls) represent the flexibility present in the system. In Section 8.4.1, locally installed energy storage devices are used for load shaping. Here, we are particularly interested in reducing fluctuations in network-wide energy consumption. To this end, we first recapitulate the basic model of a network of Residential Energy Systems (RESs) introduced and extended in [19, 41]. Then, in Section 8.4.2 controllable loads are added to further increase the flexibility with respect to the demand patterns of the RESs. Here, we rigorously prove global convergence of a setting (very) similar to the one presented in [23]. Last, in Section 8.4.3, the model is extended to additionally optimize the energy exchange between individual microgrids (MGs) coupled through transmission lines.

In the simulations, we concentrate on the performance of the overall network instead of individual performance measures. We thus set $\alpha = 1$, i.e., we use the coupling stage costs $\hat{\ell}$ (or $\bar{\ell}$, respectively) and neglect the individual stage costs ℓ_i, $i \in \mathbb{N}_{\mathscr{I}}$. However, the additional consideration of convex stage costs is straightforward. The numerical simulations are based on a dataset provided by the electricity distribution company Ausgrid and publicly available online [42]. From the dataset consisting of data collected from 300 residential customers with a resolution of half-hour windows we use the information of power consumption and power generation using solar photovoltaic (PV) panels and the information of controllable loads, which is available for some customers. The power consumption and the power generation is combined to obtain the power demand of a single customer at a particular time.

8.4.1 The Basic Model and Its MPC Formulation

Several models describing RESs have been introduced in the literature. We focus on extensions to the model discussed in [19, 41].

The dynamics of the ith RES, $i \in [1 : \mathscr{I}]$, are described by

$$x_i(k+1) = f_i(x_i(k), u_i(k)) \quad = \varrho_i x_i(k) + T(\beta_i u_i^+(k) + u_i^-(k)) \qquad (8.35)$$

$$z_i(k) = h_i(k, x_i(k), u_i(k)) = w_i(k) + u_i^+(k) + \gamma_i u_i^-(k), \qquad (8.36)$$

compare (8.9) and (8.10) in Section 8.2. In the particular setting of an electricity grid, the state $x_i \in \mathbb{R}$ represents the state of charge of a battery in kWh, containing

local information of the ith RES and the variable $z_i \in \mathbb{R}$ denotes the power demand in kW of system i, which needs to be shared with or communicated to the CE. The local power demand z_i depends on the net consumption $w_i \in \mathbb{R}$ in kW, i.e., the energy demand minus the generation of solar PV panels, and can be manipulated by charging or discharging the storage device using the input signals $u_i = (u_i^+, u_i^-)^T \in \mathbb{R}^2$ (in kW). The net consumption $w_i(k)$, which is assumed to be a known exogenous signal for all $i \in [1 : \mathscr{I}]$ and for all $n \in [k : k + N - 1]$, causes the time dependency of the function h in (8.36). Hence, the prediction horizon corresponds to the time interval, on which *reliable* data is available, see, e.g. [41] for the impact of forecast errors. The additional parameters in (8.35)–(8.36) are constants, where $T > 0$ represents the length of a sampling interval in hours (h), while $\varrho_i, \beta_i, \gamma_i \in (0, 1]$ are used to model losses due to energy transformation.

Additionally, for constants $C_i, \underline{u}_i, \overline{u}_i \in \mathbb{R}_{>0}$, $i \in [1 : \mathscr{I}]$, constraints on the state of charge of the storage device and maximal charging and discharging rates are defined in [19, 41], which result in the state constraints

$$x_i(k) \in \mathbb{X}_i = \{x_i(k) \in \mathbb{R} | 0 \leq x_i(k) \leq C_i\} \tag{8.37}$$

and the input constraints

$$u_i(k) = \begin{pmatrix} u_i^+(k) \\ u_i^-(k) \end{pmatrix} \in \mathbb{U}_i = \left\{ u_i \in \mathbb{R}^2 \left| \begin{array}{ccc} -\underline{u}_i \leq & u_i^- & \leq 0 \\ 0 \leq & u_i^+ & \leq \overline{u}_i \\ 0 \leq & \frac{u_i^-}{-\underline{u}_i} + \frac{u_i^+}{\overline{u}_i} & \leq 1 \end{array} \right. \right\} \tag{8.38}$$

for all $i \in [1 : \mathscr{I}]$ and for all $k \in \mathbb{N}$. Note that in this setting neither the state constraints $\mathbb{X}_i$ nor the input constraints $\mathbb{U}_i$ are time dependent.

In [40, 41] the objective function in the MPC closed-loop formulation is defined in such a way, that for all $k \in \mathbb{N}$ in the prediction horizon, the deviation of the aggregated power demand $\sum_{i=1}^{\mathscr{I}} z_i(k)$ from the aggregated net consumption computed over the prediction horizon is minimized. Here, we use a slightly different approach to be able to use the definition (8.11) of the stage costs. For a given prediction horizon $N \in \mathbb{N}$, we denote the average net consumption over the prediction horizon of a single RES $i \in [1 : \mathscr{I}]$ by

$$\zeta_i(k) = \begin{cases} \frac{1}{k+1} \sum_{n=0}^{k} w_i(n), & \text{if } k \leq N - 1, \\ \frac{1}{N} \sum_{n=k-N+1}^{k} w_i(n), & \text{if } k \geq N - 1. \end{cases} \tag{8.39}$$

For the first $N - 1$ time steps, where the past data of $w_i(k)$, for $k < 0$ is not available, only the average over the available data is used. Analogously the average net consumption of all the systems is defined as

$$\overline{\zeta}(k) = \frac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \zeta_i(k) \tag{8.40}$$

for all $k \in \mathbb{N}$. With this definition the coupled stage costs, tracking the average net consumption of the RESs can be written in the form

$$\bar{\ell}(k, \tfrac{1}{\mathscr{I}} \textstyle\sum_{i=1}^{\mathscr{I}} z_i(k)) = \left(\bar{\zeta}(k) - \tfrac{1}{\mathscr{I}} \textstyle\sum_{i=1}^{\mathscr{I}} z_i(k) \right)^2 \tag{8.41}$$

and the overall objective function of the OCP (8.12) in the distributed MPC Algorithm 1 at time $k \in \mathbb{N}$ is defined by

$$\phi(\mathbf{y}; k) = \textstyle\sum_{n=k}^{k+N-1} \left(\bar{\zeta}(n) - \tfrac{1}{\mathscr{I}} \textstyle\sum_{i=1}^{\mathscr{I}} z_i(n) \right)^2. \tag{8.42}$$

Here, the local terms in the objective function are not present since the weighting parameter is set to $\alpha = 1$. The objective function is convex but not strictly or strongly convex. The sets $\mathbb{D}_i(k)$, $i \in [1 : \mathscr{I}]$ and $k \in \mathbb{N}$, are defined by the initial state of charge, the system dynamics (8.35)–(8.36) and the constraints (8.37)–(8.38), see (8.17). Since the dynamics and the constraints are linear, $\mathbb{D}_i(k)$ is convex and closed. Additionally, compactness of $\mathbb{D}_i(k)$ can be concluded from the fact that $\mathbb{U}_i$ and $\mathbb{X}_i$, $i \in [1 : \mathscr{I}]$ are compact and the boundedness of $\mathbb{U}_i$ implicitly limits the power demand z_i for all $i \in [1 : \mathscr{I}]$. Thus, the distributed optimization Algorithm 3 can be used to compute an optimal solution of the OCP embedded in the MPC Algorithm 1.

Numerical simulations

For the numerical simulations throughout this section a setting of $\mathscr{I} = 100$ RESs is used. The parameter T is set to $T = 0.5$, representing half-hour windows, and the energy demand is predicted for 24 h into the future, i.e., we set $N = 48$. Additionally the constants $\underline{u}_i = \overline{u}_i = 0.5$ are fixed for the maximal discharging/charging rates for all $i \in [1 : \mathscr{I}]$. The constants modeling the losses are set to $\varrho_i = 0.99$ and $\beta_i = \gamma_i = 0.95$ for all $i \in [1 : \mathscr{I}]$. The battery capacities of the storage devices are set to $C_i = 2$ kWh with initial state of charge $x_i(0) = 0.5$ kWh for all $i \in [1 : \mathscr{I}]$. For the ADMM Algorithm 3 embedded in the MPC scheme the parameter ρ is set to $\rho = 0.1$.

The closed-loop results of the MPC Algorithm 1 over a simulation length of one week (i.e., $k = 0, \ldots, 335$) are visualized in Figure 8.1.

Due to the definition of the reference value $\bar{\zeta}(k)$ in (8.40), $\bar{\zeta}(k)$ takes 24 h before it becomes a reliable reference value which only changes slowly over time (see the green line in the Figure 8.1, top). In addition to the reference value, Figure 8.1 (top) visualizes the uncontrolled average power demand $\tfrac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} w_i(k)$ (black) and the optimized average power demand $\tfrac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} z_i(k)$ (blue). The storage devices help to reduce the peaks in the average power demand significantly. The deviation of the uncontrolled power demand (black) and the controlled power demand (blue) from the reference value is shown in Figure 8.1 (middle). Even though the peaks are reduced, the MPC algorithm is not able to track the reference value perfectly due to the maximal capacity of the storage devices and due to the bounds on the charging/discharging rates. This can be observed in the last plot in Figure 8.1, where the average State of Charge (SOC) of the storage devices is visualized.
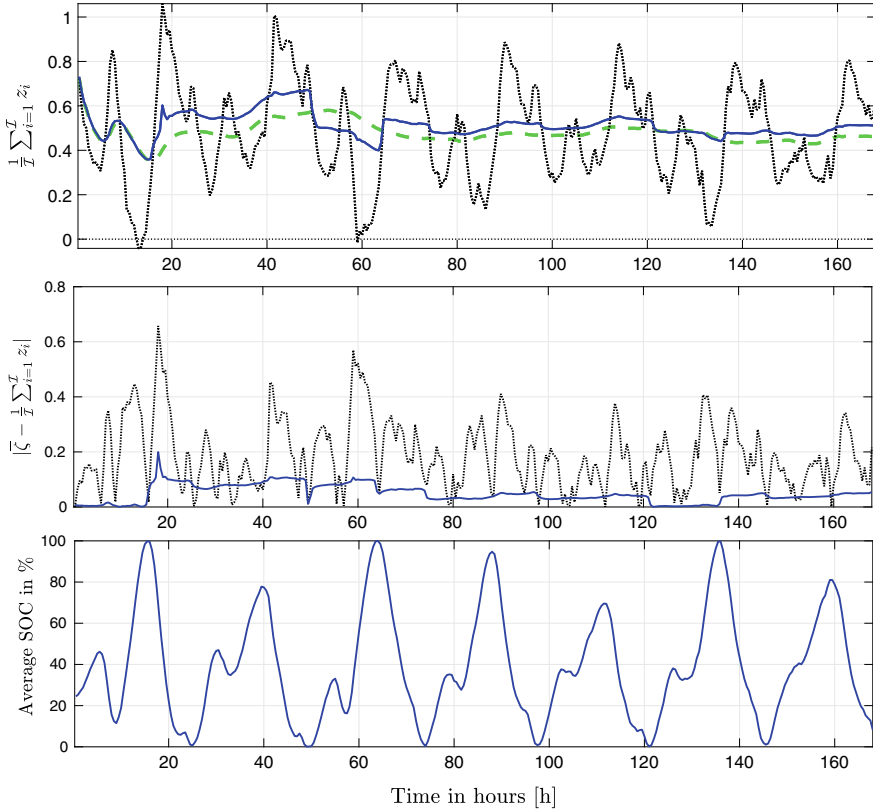
Fig. 8.1 Closed-loop results of the MPC Algorithm 1 optimizing the use of storage devices. The first figure shows the reference value $\bar{\zeta}(k)$ (green) as well as the uncontrolled (black) and controlled (blue) average power demand, $\frac{1}{\mathscr{I}}\sum_{i=1}^{\mathscr{I}} w_i(k)$ and $\frac{1}{\mathscr{I}}\sum_{i=1}^{\mathscr{I}} z_i(k)$, respectively. The second plot shows the deviation of the uncontrolled (black) and the controlled (blue) average power demand from the reference value, while the last figure visualizes the average State of Charge (SOC) of the storage devices

8.4.2 Storage Devices and Controllable Loads

In this section, we establish an additional degree of freedom in the model of the electricity network, i.e., we consider the concept of controllable loads introduced for this particular setting in [23]. To this end, the net consumption is split into two parts: the static load $w_i(k)$ and the controllable load $w_i^c(k)$. While we cannot influence the static load, controllable load can be shifted in time. To schedule the controllable load we extend the control input $u_i = (u_i^+, u_i^-, u_i^c)^T \in \mathbb{R}^3$ and modify the dynamics (8.36) to

$$z_i(k) = w_i(k) + u_i^+(k) + \gamma_i u_i^-(k) + u_i^c(k) \tag{8.43}$$

for all $i \in [1 : \mathscr{I}]$. Here, $u_i^c(k)$ in kW represents the amount of controllable load scheduled at time k.

We make the assumption that controllable load becomes available at a certain time $j \leq k$ and then needs to be scheduled until (its deadline at) time k, i.e., $w_i^c(k + \overline{N} - 1)$, $k \in \mathbb{Z}_{\geq 1 - \overline{N}}$, has to be planned during the time interval from $\max\{0, k\}$ to $k + \overline{N} - 1$ for a given $\overline{N} \in \mathbb{N}$. To this end, the time-varying constraints

$$\sum_{j=0}^{k} w_i^c(j) - \sum_{j=0}^{k-1} u_i^c(j) \quad \leq \quad u_i^c(k) \quad \leq \quad \sum_{j=0}^{k+\overline{N}-1} w_i^c(j) - \sum_{j=0}^{k-1} u_i^c(j) \qquad (8.44)$$

are introduced. Further, we assume that only a certain amount of the controllable load can be scheduled during one time step, which leads to

$$0 \quad \leq \quad u_i^c(k) \quad \leq \quad \overline{w}_i^c \qquad (8.45)$$

for a constant $\overline{w}_i^c \geq 0$, which is assumed to be chosen such that (8.44) can be fulfilled simultaneously. Note that in (8.44), at a fixed time instant $\overline{k} \in \mathbb{N}_0$, $u_i^c(k)$ for $k \geq \overline{k}$ represent control variables which need to be optimized, whereas $u_i^c(k)$ for $k < \overline{k}$ are constants which have been fixed at previous time steps. To be able to update the input constraints (8.38) capturing the controllable loads, we rewrite the constraints (8.44) as time-dependent upper and lower bounds

$$\lambda_i^q(k) := \sum_{j=0}^{k+q} w_i^c(j) - \sum_{j=0}^{k-1} u_i^c(j) \leq \sum_{j=k}^{k+q} u_i^c(j) \qquad (8.46a)$$

$$\Lambda_i^q(k) := \sum_{j=0}^{k+\min\{q+\overline{N}, N\}-1} w_i^c(j) - \sum_{j=0}^{k-1} u_i^c(j) \geq \sum_{j=k}^{k+q} u_i^c(j) \qquad (8.46b)$$

for the input variables u_i^c, for $i \in [1 : N]$, $k \in \mathbb{N}$ and over the prediction horizon $q \in [0 : N - 1]$. Note that the time dependency is due to the time-dependent controllable loads $w_i^c(k)$, $k \in \mathbb{N}$, as well as the decisions made by the controller to define $u_i^c(j)$ for $j < k$ for all $i \in \mathbb{N}_{\mathscr{I}}$. Since we assume that the system dynamics and in particular the load and the controllable load can only be estimated over the prediction horizon, the first sum in (8.46b) stops at the increment $k + \min\{q + \overline{N}, N\} - 1$.

Due to the linearity of the dynamics (8.43) and the constraints (8.45)–(8.46), convexity and compactness of the (now time-dependent) sets $\mathbb{D}_i(k)$, $i \in [1 : \mathscr{I}]$, $k \in \mathbb{N}$, follows the same arguments as in Section 8.4.1. For the stage costs, we additionally have to take the controllable loads into account in the computation of the average net consumption, i.e., $\zeta_i(k)$, $i \in \mathbb{N}_{\mathscr{I}}$, is defined as

$$\zeta_i(k) = \begin{cases} \frac{1}{k+1} \sum_{n=0}^{k} w_i(n) + w_i^c(n), & \text{if } k \leq N-1, \\ \frac{1}{N} \sum_{n=k-N+1}^{k} w_i(n) + w_i^c(n), & \text{if } k \geq N-1 \end{cases} \qquad (8.47)$$

in this setting.

Numerical simulations

In addition to the parameters used in the setting without controllable loads we set $\overline{N} = 12$, which represents a six hour time window for scheduling the controllable loads, and bound the controllable load for each RES $i \in [1 : \mathscr{I}]$ at time $k \in \mathbb{N}_0$ by $w_i^c(k) \leq \overline{w}_i^c = 1.25$ [kW].

The results comparing the setting with and without controllable loads can be found in Figure 8.2. Here, in addition to the results shown in Figure 8.1, the simulations including controllable loads are visualized in red.

The additional degree of freedom in the controllable loads clearly improves the performance of the MPC scheme where only storage devices are used. The deviation of the average power demand from the reference signal is hardly visible at most of the time steps in the simulation. To obtain a perfect tracking, however, i.e., to obtain $|\overline{\zeta}(k) - \frac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} z_i(k)| = 0$ for all $k = 0, 1, \dots, 335$, either the capacity of the storage devices or the percentage of controllable load from the overall load needs to be increased.

8.4.3   Optimal Operation of Coupled Microgrids

So far in this paper, and also in preceding publications, we have concentrated on the optimal operation of a single electricity network. Here, we extend these results to the optimal operation of Ξ microgrids (MGs), $\Xi \in \mathbb{N}$, coupled through a network of transmission lines as visualized in Figure 8.3, for example. Here, a network of $\Xi = 4$ MGs is shown where MG$_1$ is only connected to MG$_2$, and MG$_2$, MG$_3$ and MG$_4$ are fully connected through transmission lines.

The individual MGs can be defined as discussed in Section 8.4.1 without controllable loads or as in Section 8.4.2 with controllable loads. We use $\mathscr{I}_\kappa \in \mathbb{N}$, to denote the number of RESs in MG$_\kappa$, $\kappa \in [1 : \Xi]$. In the case where the MGs are not connected a straightforward extension of the stage costs defined in (8.41) is to consider the costs

$$\hat{\ell}(k, (z_1, \dots, z_\mathscr{I})^T(k)) = \sum_{\kappa=1}^{\Xi} \left(\mathscr{I}_\kappa \overline{\zeta}_\kappa(k) - \sum_{i=1}^{\mathscr{I}_\kappa} z_{\kappa_i}(k) \right)^2, \qquad (8.48)$$

where in contrast to (8.41) the deviation of the average net consumption $\overline{\zeta}_\kappa(k)$, $\kappa \in \mathbb{N}_\Xi$, in the MGs is penalized. This means, $\overline{\zeta}_\kappa(k)$ is defined as

$$\overline{\zeta}_\kappa(k) = \frac{1}{\mathscr{I}_\kappa} \sum_{i=1}^{\mathscr{I}_\kappa} \zeta_{\kappa_i}(k). \qquad (8.49)$$
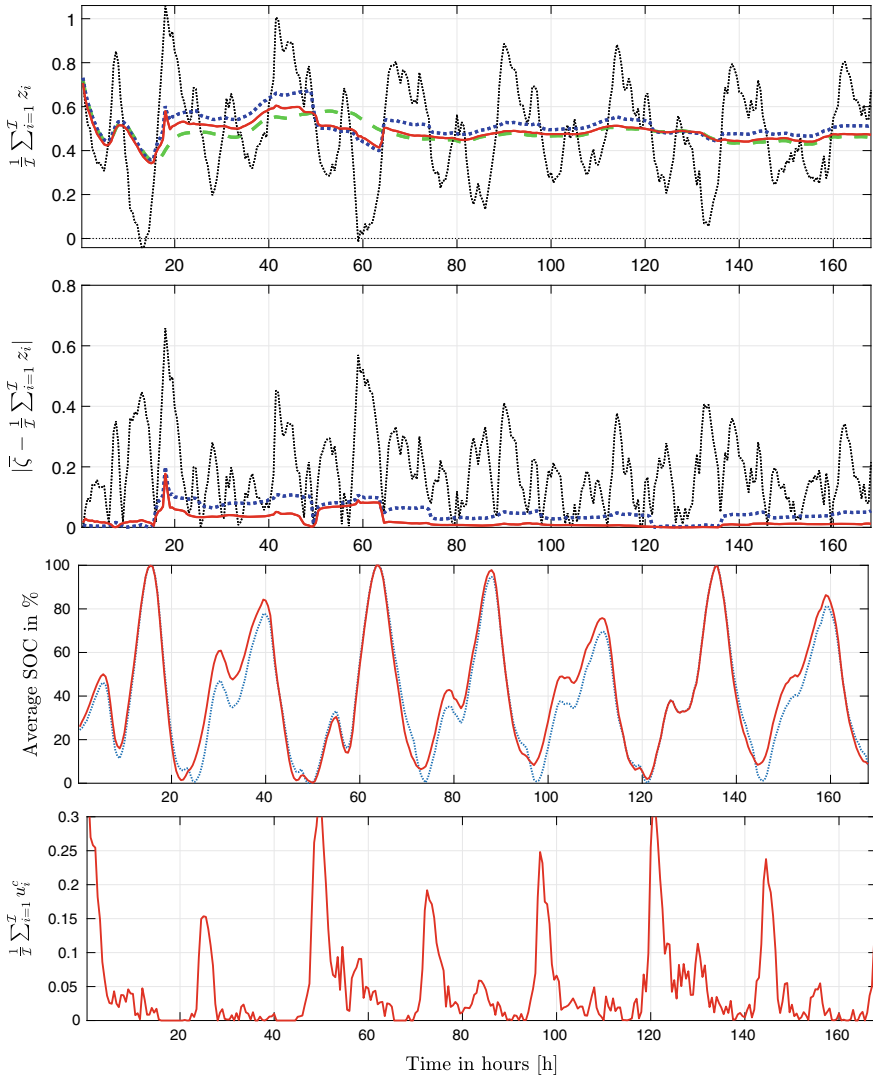
Fig. 8.2 Impact of controllable loads in the closed-loop simulation of Algorithm 1. Compared to Figure 8.1 additionally the closed-loop results with controllable loads are shown in red. The last plot shows the extended average control input $\frac{1}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} u_i^c(k), k \in [0 : 335]$
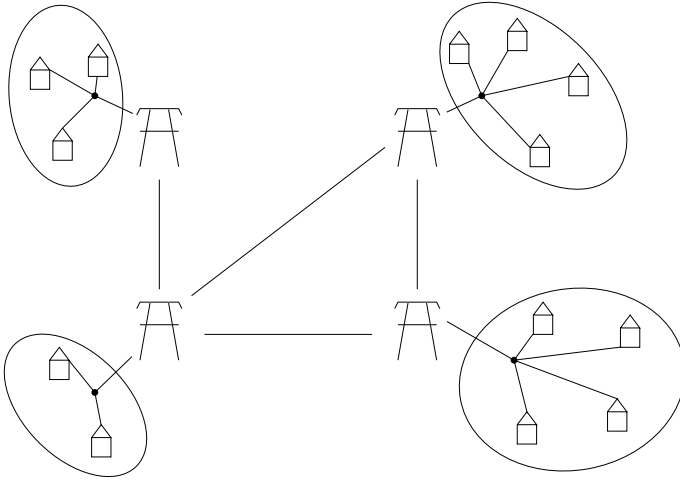
Fig. 8.3 Example of four MGs coupled through transmission lines. Each MG consists of a number of $\mathscr{I}_\kappa \in \mathbb{N}, \kappa = 1, 2, 3, 4$, RESs

We are interested in the case where the MGs are coupled via transmission lines. In particular, we want to investigate whether the MGs can benefit from an energy exchange even if the energy exchange involves losses. We thus consider stage costs of the form

$$
\hat{\ell}(k, (z_1, \ldots, z_{\mathscr{I}})^T(k), \delta(k)) = \sum_{\kappa=1}^{\Xi} \left(\mathscr{I}_\kappa \overline{\zeta}_\kappa(k) - \sum_{\nu=1}^{\Xi} (\delta_{\kappa,\nu}(k)\eta_{\kappa,\nu}) \sum_{i=1}^{\mathscr{I}_\nu} z_{i_\nu}(k) \right)^2
$$

$$
(8.50)
$$

with additional variables $\delta(k) \in [0, 1]^{\Xi \times \Xi}$ and constants $\eta \in [0, 1]^{\Xi \times \Xi}$. The matrix entry $\delta_{\kappa,\nu}(k), \kappa, \nu \in [1 : \Xi]$, represents the fraction of the power demand of MG_κ at time k, which is used to manipulate the average $\overline{\zeta}_\nu(k)$ of MG_ν. To define $\delta_{\kappa,\nu}(k)$ as a fraction of the overall power demand of MG_κ the linear constraints

$$
\sum_{\nu=1}^{\Xi} \delta_{\kappa,\nu}(k) = 1 \quad \forall \kappa \in [1 : \Xi] \tag{8.51}
$$

are introduced. Moreover, to ensure that power can be exchanged over a transmission line only in one direction at a fixed time index $k \in \mathbb{N}$ we use the nonlinear constraints

$$
\delta_{\kappa,\nu}(k) \cdot \delta_{\nu,\kappa}(k) \leq 0 \quad \forall \kappa, \nu \in [1 : \Xi], \ \kappa \neq \nu. \tag{8.52}
$$

The matrix η is assumed to be symmetric, i.e., $\eta_{\kappa,\nu} = \eta_{\nu,\kappa}$ for all $\kappa, \nu \in [1 : \Xi]$, where a *zero-entry* ($\eta_{\kappa,\nu} = \eta_{\nu,\kappa} = 0$) indicates that no transmission line between MG_ν and MG_κ exists and values $\eta_{\kappa,\nu} \in (0, 1)$ correspond to losses. In addition, we

set $\eta_{\kappa,\kappa} = 1$, $\kappa \in [1 : \Xi]$. For the MGs visualized in Figure 8.3 the constants η can be defined as

$$\eta^{\text{loss}} = \begin{pmatrix} 1.0 & 0.0 & 0.8 & 0.0 \\ 0.0 & 1.0 & 0.7 & 0.8 \\ 0.8 & 0.7 & 1.0 & 0.9 \\ 0.0 & 0.8 & 0.9 & 1.0 \end{pmatrix} \quad \text{and} \quad \eta^{\text{no loss}} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} \qquad (8.53)$$

with losses and without losses, for example.

To handle the additional set of variables δ in the MPC closed loop we extend Algorithm 1. Note that the coupling of the variables δ and $(z_1, \ldots, z_{\mathscr{I}})$ leads to a nonconvex objective function. Additionally, the constraints (8.52) are nonlinear and non-convex. Hence, δ is only optimized in the MPC algorithm, but not in the distributed optimization conducted in the second Step (2) of the MPC algorithm.

Distributed optimization of the energy exchange

Using the definitions introduced in Section 8.3 we define the objective function

$$\hat{\phi}(\mathbf{z}, \delta; k) = \sum_{n=k}^{k+N-1} \hat{\ell}(k, \mathbf{z}(k), \delta(k)) \qquad (8.54)$$

with $\delta \in \mathbb{R}^{\Xi \times \Xi \times N}$. For the constraints we define the set

$$\Delta = \left\{ \delta \in [0, 1]^{\Xi \times \Xi \times N} \middle| \begin{array}{c} \sum_{\nu=1}^{\Xi} \delta_{\kappa,\nu}(n) = 1 \ \forall \kappa \in \Xi \\ \delta_{\kappa,\nu}(n) \cdot \delta_{\nu,\kappa}(n) \leq 0 \ \forall \kappa, \nu \in [1 : \Xi], \kappa \neq \nu \\ \forall n \in [k : k + N - 1] \end{array} \right\}. \qquad (8.55)$$

Hence, the OCP involved in the MPC algorithm at time $k \in \mathbb{N}$ is defined as

$$\min_{\substack{\mathbf{y} \in \mathbb{D}(k) \\ \delta \in \Delta}} \alpha \hat{\phi}(\mathbf{z}, \delta; k) + \frac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i; k). \qquad (8.56)$$

As already pointed out, due to the additional variables δ, the distributed optimization scheme (8.33) is not applicable in the current form. Thus, we update the variables δ only once at every time step $k \in \mathbb{N}$ within the MPC algorithm and keep δ fixed in the iterates of the distributed optimization algorithm. The corresponding MPC algorithm is given in Algorithm 4.

In (8.59), δ is updated for the next time step using the ideas discussed in Section 8.3.3 and $I \in \mathbb{R}^{\Xi \times \Xi}$ denotes the identity matrix. Optimization problem (8.58) is not convex and thus it is not guaranteed, that an optimal solution is found at every time step. However, observe that the optimization problem (8.58) does not involve a coupling over the prediction horizon and $\delta(n), n \in [k : k + N - 1]$ can be computed independently, leading to N optimization problems, where the number of unknowns is upper bounded by Ξ^2. Even though optimality with respect to the energy exchange cannot be shown in our setting, simulation results indicate the potential benefit of the additional Step (3) in Algorithm 4.

Algorithm 4 Distributed MPC for energy exchange

Input: Time horizon $N \in \mathbb{N}$, number of MGs $\Xi \in \mathbb{N}$, transmission grid parameters $\nu \in [0, 1]^{\Xi \times \Xi}$, number of systems $\mathscr{I}_\kappa \in \mathbb{N}$ in MG_κ, $\kappa \in [1 : \Xi]$, satisfying $\sum_{\kappa=1}^{\Xi} \mathscr{I}_\kappa = \mathscr{I}$, stage costs $\hat{\ell}$, ℓ_i, $i \in [1 : \mathscr{I}]$, and a weighting parameter $\alpha \in [0, 1]$.

Initialization: Set $k = 0$.

Main loop: For $k \in \mathbb{N}_0$

(1) For all $i \in [1 : \mathscr{I}]$, measure the current states $\hat{x}_i := x_i(k)$.

(2) Solve the OCP (8.12) for fixed δ to obtain minimizing sequences

$$\mathbf{u}_i^\star = (u_i^\star(k), \ldots, u_i^\star(k + N - 1))^T \quad \forall i \in [1 : \mathscr{I}]. \tag{8.57}$$

(3) Obtain the optimal energy exchange δ^* by solving the optimization problem

$$\delta^* \in \underset{\delta \in \Delta}{\arg\min} \, \alpha \hat{\phi}(\mathbf{z}^*, \delta; k) + \frac{1-\alpha}{\mathscr{I}} \sum_{i=1}^{\mathscr{I}} \phi_i(\mathbf{y}_i^*; k). \tag{8.58}$$

(4) For all $i \in [1 : \mathscr{I}]$, implement $\mu_i(k, \hat{x}_i) := u_i^\star(k)$ and $\delta^*(k)$, define

$$\delta^+ = (\delta^*(k + 1) \times \cdots \times \delta^*(k + N - 1) \times I), \tag{8.59}$$

shift the horizon forward in time, i.e., set $k = k + 1$, and go to Step (1).

The power exchange of coupled MGs is also discussed in [43]. In [43] however, the power flow over the transmission lines results from linearized DC power flow equations and cannot be optimized separately. In contrast, Algorithm 4 includes the optimization of the power exchange through the additional variable δ.

Numerical simulations

To show the numerical properties of Algorithm 4 we consider a network of $\Xi = 4$ MGs visualized in Figure 8.3. Additionally, we consider the set of parameters η defined in (8.53) with and without losses. The individual MGs consist of $\mathscr{I}_1 = 20$, $\mathscr{I}_2 = \mathscr{I}_3 = 25$ and $\mathscr{I}_4 = 30$ RESs. Each RES is defined through the parameters introduced in Section 8.4.1.

The deviation of the power demand of the closed-loop solution from the reference values in the individual MGs and for the different settings is visualized in Figures 8.4 and 8.5. Figure 8.4 compares the uncontrolled power demand with the controlled power demand without energy exchange (see Section 8.4.1). In Figure 8.5 additionally, the power exchange is taken into account. The exchange of energy without losses over the transmission lines (Figure 8.5, right) clearly improves the results of the uncoupled MGs (Figure 8.4, right). The benefit of the energy exchange with losses η^{loss} over the transmission lines is not that obvious. It can however be observed by evaluating the MPC closed-loop performance (8.5), taking the average over 336 iterations. Here, the uncontrolled setting in Figure 8.4 (left) leads to average costs of 148.66 compared to 10.60 in the controlled case without energy exchange (Figure 8.4, right). The setting with energy exchange and with losses over the transmission lines (Figure 8.5, left) decreases the average costs to 3.50, which shows the improvements compared to the uncoupled simulations. As one might expect from Figure 8.5 (right)
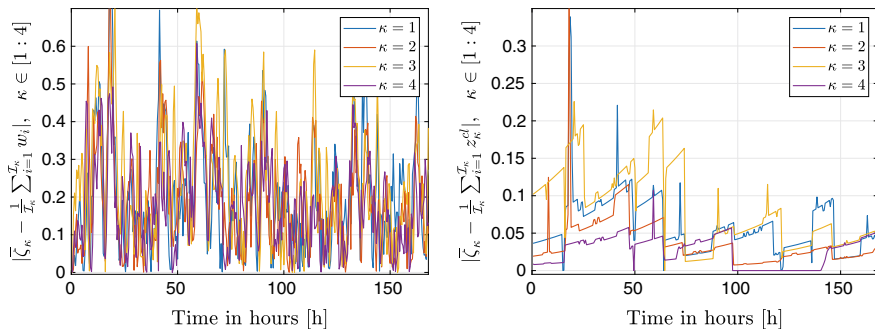
Fig. 8.4 Deviation of the uncontrolled average power demand in the MGs from the reference values in the left figure and the deviation of the closed-loop solution without power exchange (i.e., optimization with respect to (8.48)) on the right for $\kappa \in [1:4]$. Note the different scaling on the y-axis
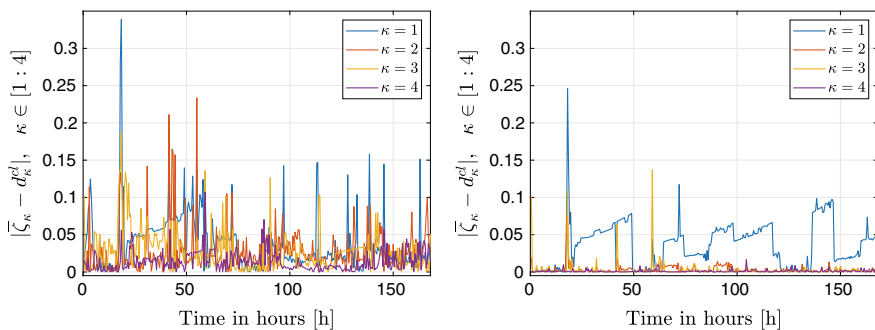
Fig. 8.5 Visualization of the deviation of the closed-loop solution from the reference value with power exchange (i.e., optimization with respect to (8.50)), with (left) and without (right) losses are visualized for $\kappa \in [1:4]$. Here, d_κ^{cl} denotes the closed-loop average power demand after the power exchange $d_\kappa^{\mathrm{cl}}(k) = \frac{1}{\mathscr{I}_\kappa} \sum_{\nu=1}^{\Xi} \delta_{\kappa,\nu}^{\mathrm{cl}}(k) \eta_{\kappa,\nu} \sum_{i=1}^{\mathscr{I}_\nu} z_{i_\nu}(k)$. The matrices η are defined in (8.53)

the setting with energy exchange and without losses outperforms the other simulation results with a value of 1.00 for the average closed-loop performance over 336 iterations.

8.5   Conclusions

In this paper, we gave a conceptual review of dual decomposition—in particular, dual ascent—and ADMM as representatives of the class of iterative distributed optimization algorithms. Furthermore, we thoroughly discussed their embedding within distributed MPC schemes. Here, we emphasized the importance to tailor the distributed optimization scheme to the particular application to ensure essential prop-

erties like scalability and plug-and-play capability. To demonstrate the effectiveness of the proposed combination, we considered an application, in which flexibilities (energy storage devices, controllable loads, and energy exchange between microgrids) were exploited for load shaving. In particular, we rigorously showed global convergence for the setting with controllable loads to extend our previous work [23] and presented a new model, which allows to (numerically) assess the potential of energy exchange between several microgrids while taking transmission losses into account.

References

1. Qin, S.J., Badgwell, T.A.: A survey of industrial model predictive control technology. Control. Eng. Pract. **11**(7), 733–764 (2003)
2. Camacho, E.F., Bordons, C.A.: Model Predictive Control in the Process Industry. Springer (2012)
3. Grüne, L., Pannek, J.: Nonlinear Model Predictive Control. Theory and Algorithms, 2nd edn. Springer, London (2017)
4. Rawlings, J.B., Mayne, D.Q., Diehl, M.M.: Model Predictive Control: Theory, Computation, and Design. Nob Hill Publishing (2017)
5. Lee, J.H.: Model predictive control: review of the three decades of development. Int. J. Control. Autom. Syst. **9**(3), 415 (2011)
6. Diehl, M., Bock, H.G., Schlöder, J.P., Findeisen, R., Nagy, Z., Allgöwer, F.: Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. J. Process. Control. **12**(4), 577–585 (2002)
7. Wang, Y., Boyd, S.: Fast model predictive control using online optimization. IEEE Trans. Control. Syst. Technol. **18**(2), 267–278 (2010)
8. Houska, B., Frasch, J., Diehl, M.: An augmented Lagrangian based algorithm for distributed nonconvex optimization. SIAM J. Optim. **26**(2), 1101–1127 (2016)
9. Benders, J.F.: Partitioning procedures for solving mixed-variables programming problems. Numer. Math. **4**(1), 238–252 (1962)
10. Bertsekas, D.P., Tsitsiklis, J.N.: Parallel and Distributed Computation: Numerical Methods. Athena Scientific, Belmont, MA, USA (1989)
11. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn. **3**(1), 1–122 (2011)
12. Stewart, B.T., Venkat, A.N., Rawlings, J.B., Wright, S.J., Pannocchia, G.: Cooperative distributed model predictive control. Syst. Control. Lett. **59**(8), 460–469 (2010)
13. Venkat, A.N., Hiskens, I.A., Rawlings, J.B., Wright, S.J.: Distributed MPC strategies with application to power system automatic generation control. IEEE Trans. Control. Syst. Technol. **16**(6), 1192–1206 (2008)
14. Giselsson, P., Doan, M.D., Keviczky, T., De Schutter, B., Rantzer, A.: Accelerated gradient methods and dual decomposition in distributed model predictive control. Automatica **49**(3), 829–833 (2013)
15. Giselsson, P., Rantzer, A.: Distributed model predictive control with suboptimality and stability guarantees. In: Proceedings of 49th IEEE Conference on Decision and Control (CDC), pp. 7272–7277 (2010)

16. Giselsson, P., Rantzer, A.: On feasibility, stability and performance in distributed model predictive control. IEEE Trans. Autom. Control. **59**(4), 1031–1036 (2014)
17. Christofides, P.D., Scattolini, R., de la Pena, D.M., Liu, J.: Distributed model predictive control: a tutorial review and future research directions. Comput. Chem. Eng. **51**, 21–41 (2013)
18. Maestre, J.M., Negenborn, R.R.: Distributed Model Predictive Control Made Easy, vol. 69. Springer (2014)
19. Braun, P., Grüne, L., Kellett, C.M., Weller, S.R., Worthmann, K.: Towards price-based predictive control of small-scale electricity network. Int. J. Control. (2017). https://doi.org/10.1080/00207179.2017.1339329
20. Chatzipanagiotis, N., Dentcheva, D., Zavlanos, M.M.: An augmented Lagrangian method for distributed optimization. Math. Program. **152**(1–2), 405–434 (2015)
21. Yi, P., Hong, Y., Liu, F.: Distributed gradient algorithm for constrained optimization with application to load sharing in power systems. Syst. Control. Lett. **83**, 45–52 (2015)
22. Rahman, H.A., Majid, M.S., Jordehi, A.R., Gan, C.K., Hassan, M.Y., Fadhl, S.O.: Operation and control strategies of integrated distributed energy resources: a review. Renew. Sustain. Energy Rev. **51**, 1412–1420 (2015)
23. Braun, P., Grüne, L., Kellett, C.M., Weller, S.R., Worthmann, K.: Model predictive control of residential energy systems using energy storage and controllable loads. In: Progress in Industrial Mathematics at ECMI 2014, pp. 617–623. Springer (2016)
24. Boccia, A., Grüne, L., Worthmann, K.: Stability and feasibility of state constrained mpc without stabilizing terminal constraints. Syst. Control Lett. **72**, 14–21 (2014)
25. Angeli, D., Amrit, R., Rawlings, J.B.: On average performance and stability of economic model predictive control. IEEE Trans. Autom. Control. **57**(7), 1615–1626 (2012)
26. Müller, M.A., Grüne, L.: Economic model predictive control without terminal constraints: optimal periodic operation. Automatica **70**(C), 128–139 (2016)
27. Damm, T., Grüne, L., Stieler, M., Worthmann, K.: An exponential turnpike theorem for dissipative discrete time optimal control problems. SIAM J. Control. Optim. **52**(3), 1935–1957 (2014)
28. Faulwasser, T., Bonvin, D.: Exact turnpike properties and economic NMPC. Eur. J. Control. **35**, 34–41 (2017)
29. Grüne, L., Pirkelmann, S.: Closed-loop performance analysis for economic model predictive control of time-varying systems. In: Proceedings of the 54th IEEE Conference on Decision and Control (CDC), pp. 5563–5569 (2017)
30. Köhler, J., Müller, M.A., Allgöwer, F.: Nonlinear reference tracking: an economic model predictive control perspective. IEEE Trans. Autom. Control. **64**(1), 254–269 (2019)
31. Raff, T., Huber, S., Nagy, Z.K., Allgöwer, F.: Nonlinear model predictive control of a four tank system: an experimental stability study. In: Proceedings of the IEEE Conference on Control Applications, pp. 237–242 (2006)
32. Müller, M.A., Worthmann, K.: Quadratic costs do not always work in MPC. Automatica **82**, 269–277 (2017)
33. Eichfelder, G.: Adaptive Scalarization Methods in Multiobjective Optimization, vol. 436. Springer (2008)
34. Grüne, L., Stieler, M.: Performance guarantees for multiobjective model predictive control. In: Proceedings of the 56th IEEE Conference on Decision and Control (CDC), pp. 5545–5550 (2017)
35. Zeilinger, M.N., Pu, Y., Riverso, S., Ferrari-Trecate, G., Jones, C.N.: Plug and play distributed model predictive control based on distributed invariance and optimization. In: Proceedings of the 52nd IEEE Conference on Decision and Control (CDC), pp. 5770–5776 (2013)
36. Lucia, S., Kögel, M., Findeisen, R.: Contract-based predictive control of distributed systems with plug and play capabilities. IFAC-PapersOnLine **48**(23), 205–211 (2015)
37. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific (1999)
38. Duchi, J.C., Agarwal, A., Wainwright, M.J.: Dual averaging for distributed optimization: convergence analysis and network scaling. IEEE Trans. Autom. Control. **57**(3), 592–606 (2012)
39. Parikh, N., Boyd, S.P.: Proximal algorithms. Found. Trends Optim. **1**(3), 123–231 (2013)

40. Braun, P., Faulwasser, T., Grüne, L., Kellett, C.M., Weller, S.R., Worthmann, K.: Hierarchical distributed ADMM for predictive control with applications in power networks. IFAC J. Syst. Control. **3**, 10–22 (2018)
41. Worthmann, K., Kellett, C.M., Braun, P., Grüne, L., Weller, S.R.: Distributed and decentralized control of residential energy systems incorporating battery storage. IEEE Trans. Smart Grid **6**(4), 1914–1923 (2015)
42. Ausgrid dataset. https://www.ausgrid.com.au/Common/About-us/Corporate-information/Data-to-share/Solar-home-electricity-data.aspx. Accessed 18 May 2018
43. Hans, C.A., Braun, P., Raisch, J., Grüne, L., Reincke-Collon, C.: Hierarchical distributed model predictive control of interconnected microgrids. IEEE Trans. Sustain. Energy **10**(1), 407–416 (2019)

Chapter 9
Coherency Estimation in Power Systems: A Koopman Operator Approach

Harold R. Chamorro, Camilo A. Ordonez, Jimmy C.-H. Peng, Francisco Gonzalez-Longatt and Vijay K. Sood

Abstract Integrating a significant amount of non-synchronous generation into power systems creates new technical challenges for transmission systems. The research and understanding of the impact of the non-synchronous generation through back-to-back Full Rated Converters' (FRCs) on power system's coherency is a matter of importance. Coherency behavior under the presence of large inclusion of non-synchronous generation requires more research, in order to understand the forming groups, after a disturbance, when the inertia is decreasing due to the decoupling. This document presents the application of the so-called *Koopman Operator* for the identification of coherent groups in power systems with the influence of non-synchronous generation. The *Koopman* Analysis clusters the coherent groups based on the measurements obtained. The visualization of the coherent groups identified allows to observe their dynamic variations according to the penetration level or fault location. The applied method of coherency identification is evaluated in the Nordic test system through gradually increasing integration of non-synchronous generations and different fault scenarios.

H. R. Chamorro (✉)
KTH Royal Institute of Technology, Teknikringen 33, Stockholm, Sweden
e-mail: hrcv@kth.se; hr.chamo@ieee.org

C. A. Ordonez
Grupo Energia Bogota, Carrera 9 # 73-44, Bogotá, Colombia
e-mail: cordonezm@geb.com.co

J. C.-H. Peng
National University of Singapore, Singapore, Singapore
e-mail: elepcj@nus.edu.sg

F. Gonzalez-Longatt
Loughborough University, Holywell Park, Epinal Way, Loughborough, UK
e-mail: f.gonzalez-longatt@lboro.ac.uk

V. K. Sood
University of Ontario Institute of Technology, Oshawa, ON L1H 7K4, Canada
e-mail: vijay.sood@uoit.ca

9.1   Introduction

The considerable environmental benefits of integrating renewable generation into the grid have encouraged several governmental policies around to world. Since the non-synchronous generation (wind, wave and solar power) is required to behave like conventional synchronous generation units, (full-scale power supply) the high voltage power electronics converters have become attractive to be selected for the integration into either the transmission or distribution power levels [1].

A wider use of non-synchronous generation relies on the use of Full Rated Converters (FRCs). This is due to them enabling multiple control features which include controlling active and reactive power [2], assuring voltage ride through capability [3] in order to deal with variable speed wind turbines [4], and adding Maximum Power Point Tracking (MPPT) algorithms [5]. However, the generation is completely decoupled from the system, and is consequently unable to contribute dynamically to the system [6, 7]. Thus, if the synthetic inertia control option is not added, the decoupling displaces synchronous machines [8].

The reduction of the inertia in the power grid has provoked a global concern by the system operators, power planners and researchers. Several reports have shown the experience with large inclusion of non-synchronous generation and the dynamical challenges during the past years [9].

Some of the main challenges which have been reported include the following:

– The first experienced challenge is the decrease in the system inertial response which affects the dynamical response [10–12].
– The second challenge due to non-synchronous integration is the impact on small signal stability. Several studies have shown that the large-scale inclusion of wind power degrades the damping of the electromechanical modes. This displace them to a different stability operation region, thereby affecting the response of the system under small and large disturbances [13, 14].

A study conducted in [15] analyses the impact of the effect of high penetration of photovoltaic (PV) on small signal stability. Due to the reduced system inertia the study result showed a reduction in the damping torques of the system. A transient and small signal stability analyses with a gradual inclusion of PV are developed in [16], the eigenvalue results show that a displacement of conventional units have a great impact on the oscillatory modes.

– The third challenge is the coherency of power systems and how it can be affected by the use of high power electronic converters. One relevant study shows that the large-scale inclusion of wind power changes the coherency of the synchronous generators coherent groups [17], however additional research is required in understanding how the non-synchronous generation affects the coherency in power systems.

The phenomenon of coherency in large interconnected power systems is presented when some generators swing together after a disturbance [18, 19]. To identify coherent groups, different methods have been studied during the last years in literature.

Time and frequency-domain methods have been used to analyse the coherency of generators [20, 21]. Partitioning-based slow coherency methods have been presented in [22, 23].

Support vector clustering is applied for coherency studies in [24], computing compact and separated clusters based on the measurements obtained.

Some approaches applying machine learning and computational intelligence techniques for coherent identification have also been investigated in [25, 26]. In [27], neural networks, as patterns recognition and classificators sets are used. In a more recent contribution [28], where neural networks are also used, a fast method for security assessment is proposed. Fuzzy clustering methods have also been applied in [29] and [30] with auto-configuring training, and c-means clustering methods, respectively.

Another related method is presented in [31], which applies maximum spanning tree partitioning, to obtain the strongest connections in the network after a disturbance.

One method is developed in [32], where the flocking agent-interaction method is applied for different scenarios including rapid clustering identification.

Digital signal processing techniques have been applied in this topic. One important contribution is presented in [33], where the wavelet transform is used to obtain the phase relations according to the signals, and is used to determine the common frequencies of the generators involved. Another important approach is given in [34], where the Hilbert-Huang transform is tracking the generator coherency instantaneously.

Coherency-based graph theory has been studied in [35] and [36], where the topological network structure is analysed. In this method the generators are clustered according to the sub-networks and cut-sets obtained.

In [17], the coherency including wind farms is studied. The determination of the coherent groups is done by the rotor angle response observation after a fault.

The *Koopman operator* has proven to be a suitable method for coherency identification in power systems [37]. Koopman Mode Analysis (KMA) provides a graphical tool based on linear transformations on Hilbert spaces to analyse (non-linear) Hamiltonian systems. This linear, infinite-dimensional operator is defined for any non-linear dynamical system [38, 39]. One important characteristic of the Koopman operator is its ability to capture the full pattern information of large complex dynamical systems like power grids. Also, in [40], an islanding method is proposed and it is shown that the Koopman operator can determine the static connectivity of a system in a similar way as graph theory does.

This document follows the theory presented in [41], and also applies the technique developed in [37] and [42]. Otherwise, the same mentioned authors, in a most recent contribution of KMA presented in [43], the authors show the application in the 2006 European Grid-wide disturbance [44]. It is demonstrated the versatility of the KMA by decomposing the power exchanges between the operative areas in order to not only observe the coupled swing dynamics, but use the KMA to diagnose the instabilities using the data obtained from a real past case [43].

The main contributions of this document are the follow ones. The first one, is the coherent groups identification including non-synchronous generations in power systems which has been barely studied in literature before, and second one, to address the notion of how the reduction of inertia by the large penetration of power converters change affects the coherency in power systems from the data. Note the Koopman Analysis requires only the measurements of the system, not the model, which makes a robust and practical graphical tool. Moreover, the analysis is illustrated in a test system under the consideration of different penetration level scenarios and fault locations illustrating gradually the coherent groups conformation.

This document is an extended version of [2, 70] and is structured as follows. In Section 9.2, the problem formulation of coherency identification in non-synchronous integration is presented. In Section 9.3, the preliminaries of the theory of the *Koopman operator*, the definition of KM and the coherency definition are reviewed. In Section 9.4, the coherency in the Nordic test system is studied under different faults. Section 9.5 presents the case studies regarding the gradual increasing of non-synchronous generation integration on the test system. Finally, the conclusions and future work of this research are given.

9.2   Problem Formulation

The identification of coherent groups is of importance for the development, deployment and implementation of control schemes to improve the system transmission capability [45]. Having identified coherent groups, it is possible to classify those generators that are oscillating together, in order to understand the dynamic behaviour of the system under disturbances, and design appropriate controllers to protect it against them [46].

Coherency identification can be used in Wide Area Monitoring Protection and Control (WAMPAC) systems [47] in different ways. For instance, controlled islanding uses the coherency identification to know how to split the network into different sub-groups, and to avoid a blackout of the complete system [48, 49].

A dynamic preventive observation of the coherent groups can confine cascading faults within smaller self-sustainable islands making the grid more robust under disturbances [50].

Self-healing schemes can also be improved through the observation of the coherent groups to protect, apply control actions and restore the system after a fault [51].

The identification of coherent groups can also be helpful to locate, design or tune Power System Stabilizers (PSS) [52, 53]. Furthermore, the design of Flexible AC Transmission Systems (FACTS) controllers can also take advantage of the coherent groups identification as shown in [54], where the feedback measurements are selected based on the dominant machines of the identified groups.

Moreover, cyber-security schemes can take advantage of the coherent identification as a tool for monitoring the dynamic changes instantaneously and upgrade the control decisions faster [55, 56]. Furthermore, it is possible to propose dynamic security assessments to make the response of the protection systems more efficient

and adaptable, and also to conceive advanced alert systems in order to avoid large area disturbances [57, 58].

Coherency identification methods can be classified into two basic groups [59]. The first one is based on linearised models of power systems [60, 61]. These methods might not be completely suitable since they do not capture the post-fault dynamics. The second group can be called measurement-based methods, because they rely on the signals obtained from the system, either off-line or on-line methods, to bring a more precise dynamic observation of the system [62].

9.2.1 Koopman Self-clustering Optimization

One of the biggest concerns in Wide Area Measurement Systems (WAMS) is the effective computing techniques and measurement technologies which allow to visualize the dynamics of the systems and use the data for short/long term power system planning [63, 64]. Koopman Mode Analysis (KMA), in its current stage of development, is capable of receiving the signals from simulation or measured data, providing the clusters of the coherent groups. Certainly, the method has the potential to be a real-time coherency method; but further research is needed. Otherwise, the computational effort of the analysis is based only in the mathematical (numerical) calculation which can be carried out by any of the processors developed in the current technological age, or it can be easily embedded in a Hardware in-the-Loop (HiL) system or even a Floating Point Gate Array (FPGA). However, these two latest aspects are not the purpose of this document.

The visualization of the method also plays an important role, e.g. from the Transmission System Operator (TSO) point of view. The display of the results should be easy to understand by the operator, who is responsible for planning the interconnection of HVDC lines or large non-synchronous generation, and determine decisions on-time, propose control strategies to prevent undesirable events, etc. The results, and the final results display obtained by the use of the KMA (phase vs. amplitude diagrams), can bring a friendly interpretation of the dynamics variations in the power systems supervised.

One of the main robust characteristic of KMA is that can handle several data (e.g from data-receivers or predicted models) providing an assessment for short/long-term futuristic power planing and update the control systems based on the analysis given [43].

Partitioning power networks (islanding) for protecting the grid of blackouts and large collapses using KMA has an important advantage, the model-free and the relying on dynamics data in the network [40].

Coherency identification using KMA belongs to the measurement-based group since it only requires sampled data [65]. Apart of the mentioned advantages, KMA has some important characteristics which makes it a powerful tool. The first, and most important one is that it deals with non-linear dynamics. Secondly, it does not need

any training process, such as neural networks or support vector machines, because this method lies on the spectrum calculation of the Koopman Operator.

The aim of this paper is to show that the KMA can be used for identification of coherent swings and generators in systems under the effect of the decoupling of full converters. In this paper, non-synchronous generation is modelled by aggregated full converter units with the same power rating as that of the replaced synchronous generation.

9.3 Koopman Operator Preliminaries

9.3.1 Koopman Operator, Eigenvalue and Mode

Consider the following dynamics described by a discrete-time non-linear system evolving on a smooth manifold [40, 66].

$$x_{k+1} = F(x_k) \tag{9.1}$$

where, $x_k \in M$ is the state variable belonging to state space M, and $F : M \to M$ is a non-linear, vector-valued function. The *Koopman Operator* is a linear operator $\mathscr{U}$ that acts on scalar-values functions on M in the following manner: for $g : M \to \mathbb{R}$, $\mathscr{U}$ maps g into a new function $\mathscr{U} g$ given by

$$\mathscr{U} g(x) = g(F(x))$$

Although the dynamical system is non-linear and evolves on a finite-dimensional space, the *Koopman operator* is linear, but infinite-dimensional. The *Koopman eigenfunctions* $\varphi_j : M \to \mathbb{C}$ and the *Koopman eigenvalues* (KEs) $\lambda_j \in \mathbb{C}$ associated to $\mathscr{U}$, are defined as

$$\mathscr{U} \varphi_j(x) = \lambda_j \varphi_j(x), \qquad j = 1, 2, \ldots$$

Consider $g : M \to \mathbb{R}^p$ a vector-valued observable. If each g_i of the components in $\boldsymbol{g}$ lies within the span of eigenfunction φ_j, then the time-evolution of observable $\boldsymbol{g}(\boldsymbol{x}_k)$ from $\boldsymbol{g}(\boldsymbol{x}_0)$ is expanded as follows:

$$\boldsymbol{g}(\boldsymbol{x}_k) = \sum_{j=1}^{\infty} U^k \varphi_j(\boldsymbol{x}_0) \boldsymbol{v}_j = \sum_{j=1}^{\infty} \lambda_j^k \varphi_j(\boldsymbol{x}_0) \boldsymbol{v}_j \tag{9.2}$$

where, $\boldsymbol{v}_j$ is the vector-valued coefficient of the decomposition and is called the j-th Koopman Mode (KM) [67]. This decomposition is based on the properties of the point spectrum of U, and the analysis based on (9.2) is called Koopman Mode Analysis (KMA). The KMA enables the extraction of single-frequency modes from

data on fully non-linear dynamics from the temporal behaviour of the corresponding KM. The phase of λ_j determines its frequency, and the magnitude determines the growth rate.

The computation of Koopman eigenvalues and KMs is effectuated by a modified version of *Arnoldi* algorithm described in [68], where it is shown that the *Ritz* value $\tilde{\lambda}_j$ and vector $\tilde{v}_j$ approximate the Koopman eigenvalue λ_j and factor $\varphi_j(x_0) v_j$ in the expansion in terms of a finite truncation.

The input of the algorithm is the $N+1$ sampled data $\{g(x_0), g(x_1), \ldots, g(x_N)\}$. The outputs are N pairs of Koopman eigenvalues and KMs. The finite sum expansion is expressed by:

$$\begin{cases} g(x_k) = \sum_{j=1}^{N} \tilde{\lambda}_j^k \tilde{v}_j & k = 0, \ldots, N-1 \\ g(x_N) = \sum_{j=1}^{N} \tilde{\lambda}_j^N \tilde{v}_j + r \end{cases} \tag{9.3}$$

where, r is a residue with the approximation error.

9.3.2 Coherency in the Koopman Mode

By denoting $\tilde{v}_{ji}$ as the i-th element of v_j, a coherent group of KMs is identified based on the amplitude coefficient $A_{ji} := |\tilde{v}_{ji}|$ and the initial phase $\alpha_{ji} := arg(\tilde{v}_{ji})$ for each mode j and observable i (e.g. rotor angle δ_i and voltage angle θ_i).

Coherency for KMs is defined in [51] as follows. For given finite N modes $\{\tilde{v}_1, \ldots, \tilde{v}_N\}$ and fixed constants $(\varepsilon_1, \varepsilon_2)$, two observables $\{g_k, g_v\}$ are called $(\varepsilon_1, \varepsilon_2)$-coherent with respect to mode j if

$$\begin{cases} |A_{j,k} - A_{j,v}| & < \varepsilon_1 \\ |\alpha_{j,k} - \alpha_{j,v}| & < \varepsilon_2 \end{cases}$$

In this case, $g(x_k)$ contains swing dynamics of synchronous machines in a power system, so it is possible to group the oscillatory components with similar amplitude A_{ji} and initial phase α_{ji} of machines to state them as coherent. This is illustrated in the Figure 9.1.

Fig. 9.1 Groups of $(\varepsilon_1, \varepsilon_2)$-coherent observables illustrated in a phase versus amplitude plot
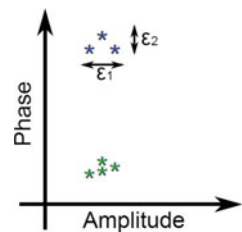
Fig. 9.2 Three sine waves
of different frequency
corresponding to (9.4). The
2.6 Hz sine wave is displaced
by a $\pi/2$ rads phase shift
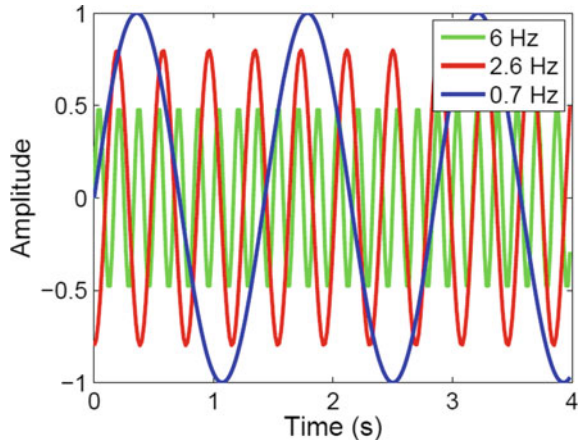
9.3.3 Illustrative Example

Let us consider a simple example of KMA. The signals shown in Figure 9.2 represent
the measurements obtained from an oscillatory signal. Notice these signals are not
coupled.

$$g_1 = 0.5sin\,(2\pi f_1 t)$$
$$g_2 = 0.8sin\,(2\pi f_2 t - \pi/2) \qquad (9.4)$$
$$g_3 = sin\,(2\pi f_3 t)$$

Here f_1, f_2, and f_3, are chosen as 6, 2.6, 0.7 Hz respectively. These signals are
depicted in Figure 9.2 over a time period of 4 s.

Now, KMA is applied with a sampling frequency of $f_s = 60$ Hz to the data.
$N + 1 = 240$ samples are acquired giving $N = 239$ modes. The modes are now
listed based on the Growth Rate (GR) which is related to the damping in case of
sampled dynamics. The five modes with the largest GRs are listed in Table 9.1.

In this manner, the sampled data has been decomposed into a set of $Ritz$ values
$\tilde{\lambda}$ and vectors v. The dominant frequencies are identified by applying KMA to the

Table 9.1 Dominant $Koopman$ modes obtained for the data shown in Figure 9.2

Mode j	Growth rate	Frequency (Hz)	Norm $\|\tilde{v}_j\|$
1	0.9997	0.68	0.0269
2	0.9960	2.66	0.2814
3	0.9951	5.94	0.0266
4	0.9925	0.39	0.5401
5	0.9913	0.13	0.1816

Fig. 9.3 The modal "dynamics" of the three dominant modes. Each of the modes has three contributions corresponding to the three measured observables (the sine waves g_1, g_2, g_3)
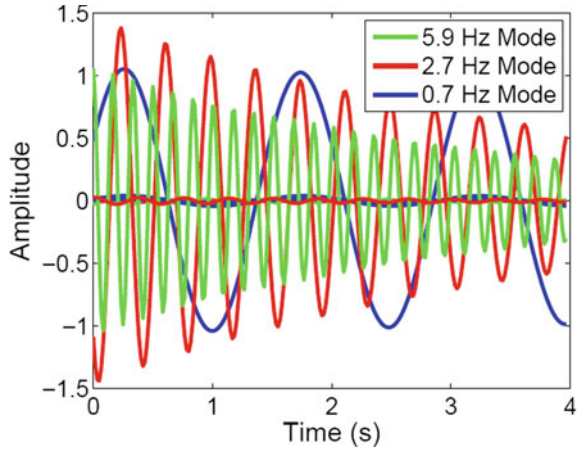
sampled data. Modal dynamics for Modes 1-3 are depicted in Figure 9.3. It is seen that for each mode, essentially only one observable (out of three) contributes (the one corresponding to the sine wave of the same frequency as the mode). The sum over all modes according to (9.2)–(9.3) reconstructs the sampled data.

9.4 Application to the Nordic System

The single-line diagram of the Nordic test system is shown in Figure 9.4. This system contains 32 high voltage buses, 20 synchronous generators with different types of generation (circled in the figure), in four geographical identified area. The *North* and *External* area are hydro-dominated while the south and central areas have a mixture of nuclear, thermal and coal power plants. Central area has the highest level consumption whereas the North area has the lowest level. The transmission system is designed for 400 kV (19 buses) with some regional systems at 220 kV (2 buses) and 130 kV (11 buses). The details of the system, such as unit rating, line data, dynamic data, and loading conditions, are given in [69]. Power System Stabilizers (PSS) have been located in the following synchronous machines: 1042, 1043, 4011, 4042, 4047, 4051, 4062, 4063.

9.4.1 Numerical Simulation

The setting of numerical simulation is as follows. The constants and power loads are the same as in [69]. All numerical simulations discussed in this paper are performed using the software DigSilent Power Factory®. Then, some disturbances (three-phase
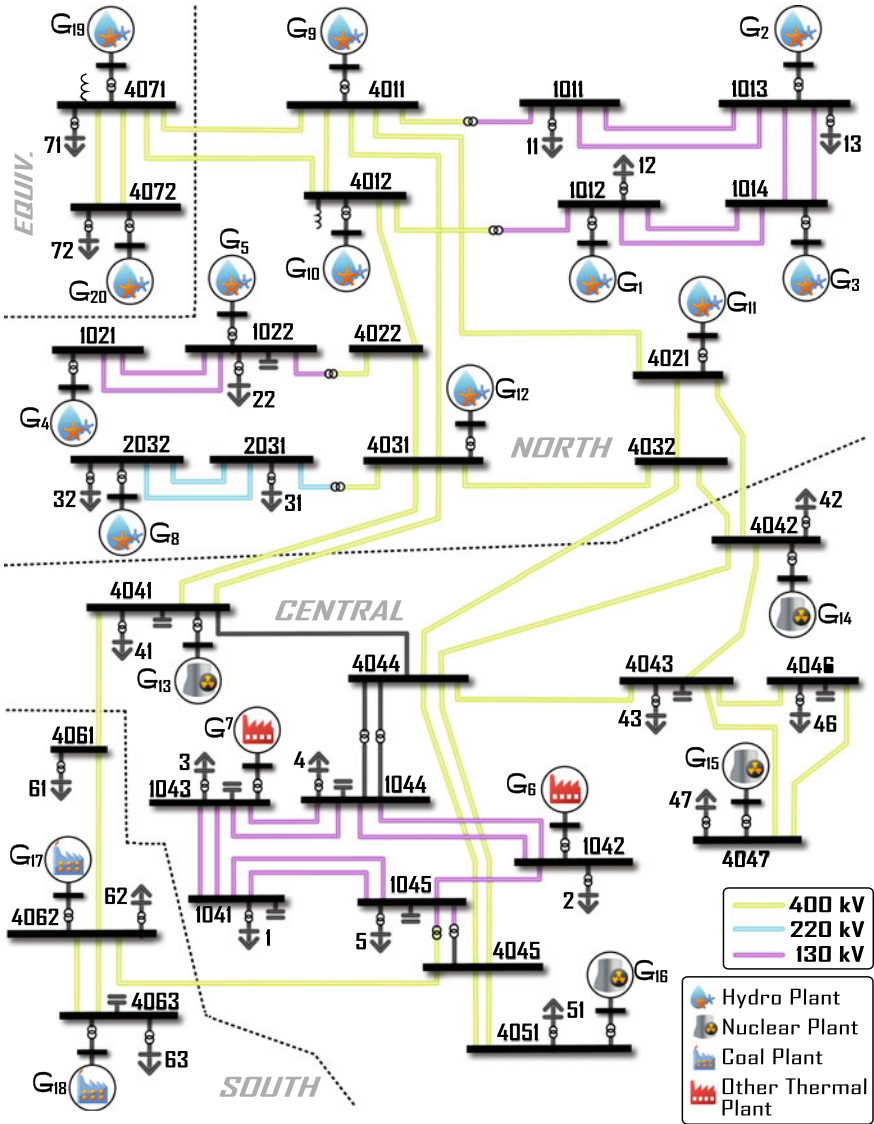
Fig. 9.4 Nordic 32 test system

Table 9.2 List of selected faults—Nordic 32-bus test system

Fault	Bus fault location	Critical clearing time (ms)
1	4047	234
2	1021	190
3	4031	180
4	4063	250
5	1012	250

short circuit faults) in the system are supposed in order to trigger the swing curves signals to analyse. The location of the faults are based on the previous study of this system presented in [70], however the Critical Clearing Time (CCT) differs due to the different dynamic settings in the system (Table 9.2).

9.4.2 Koopman Modes and Eigenvalues Analysis

KM have been computed based on the measurements obtained, which are the rotor angle δ_i of the synchronous generators. For computation, the observable $g(\delta)$ is chosen, where $\delta = [\delta_1, \ldots, \delta_7, \delta_9, \ldots, \delta_{20}]^T$. The symbol T indicates transpose in vectors. It is used on the simulation outputs obtained from the software, where the uniform sampling period $T_s = 1/(f_s)$, with $f_s = 100$ Hz and the number of samples $N = 1001$. An appropriate number of samples should be selected to capture a large quantity enough of snapshots.

Consider a fault located in the bus number B_{4047} located in the *Central* area. Figure 9.5 shows the time response of the $\Delta \delta_i$. Note that G_8 (G_{4011}) is the reference machine.

The coherent generators extraction are obtained from the KMs decomposition. In this document, the mode of interest is the one with the largest norm. The frequency related to this mode is 0.6Hz. Table 9.3 shows the numerical computation of KEs and KMs, Mode 1 to Mode 10.

The distribution of $A_{8,i}$ versus $\alpha_{8,i}$ is plotted in Figure 9.5. The circle points for generators show the different cluster groups obtained. For this mode, the phase $\alpha_{8,i}$ clusters two main generator coherent groups CG_1 (1012, 1013, 1014, 4012, 4071, 4072) and CG_2 ($G_{1043}, G_{4041}, G_{4042}, G_{4047}, G_{4051}, G_{1042}, G_{4062}, G_{4063}, G_{2032}, G_{4021}, G_{4031}, G_{1021}$ and G_{1022}) and different subgroups. Figure 9.6 show the time response of these sub-groups respectively. Observing CG_1 and CG_2 groups and the time response is possible to match which groups have positive and negative rotor angle, or in other words, the accelerating and de-accelerating groups respectively.

Following the same process, it is possible to identify the coherent groups/subgroups with the different faults stated above. Table 9.4 shows the correspond-
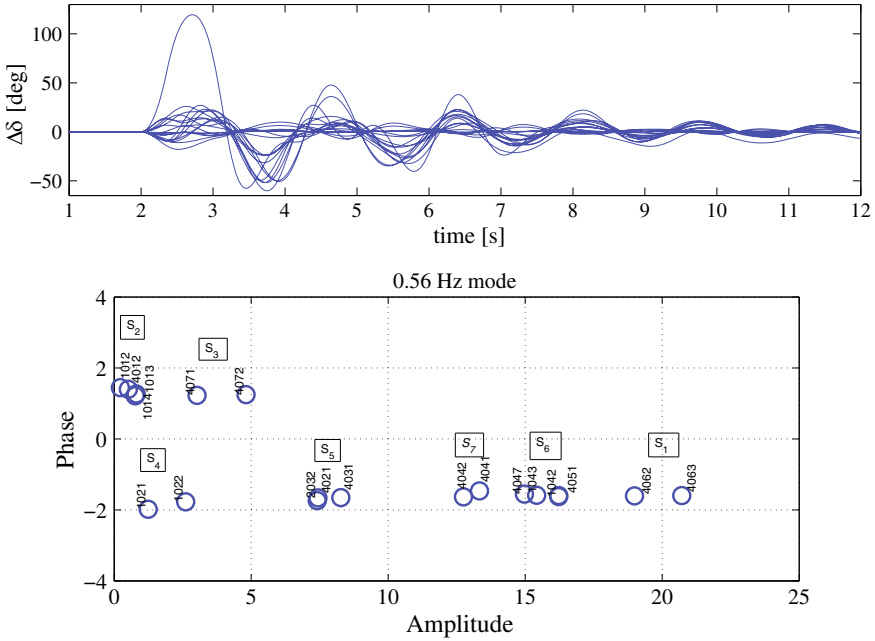
Fig. 9.5 Rotor angle time response after the fault at bus B_{4047}

Table 9.3 Result of *Koopman* modes after fault in B4047

Mode j	Growth rate	Frequency (Hz)	Norm $\|\tilde{v}_j\|$
1	1	0	115.9
2	0.9987	1.8169	0.20029
3	0.99844	2.049	0.11828
4	0.9983	2.1693	0.10667
5	0.9982	1.9336	0.18931
6	0.99809	1.6843	0.42954
7	0.99796	2.2854	0.11777
8	0.99748	0.59578	73.438
9	0.99746	0.2983	2.3481
10	0.99745	1.3594	1.6117
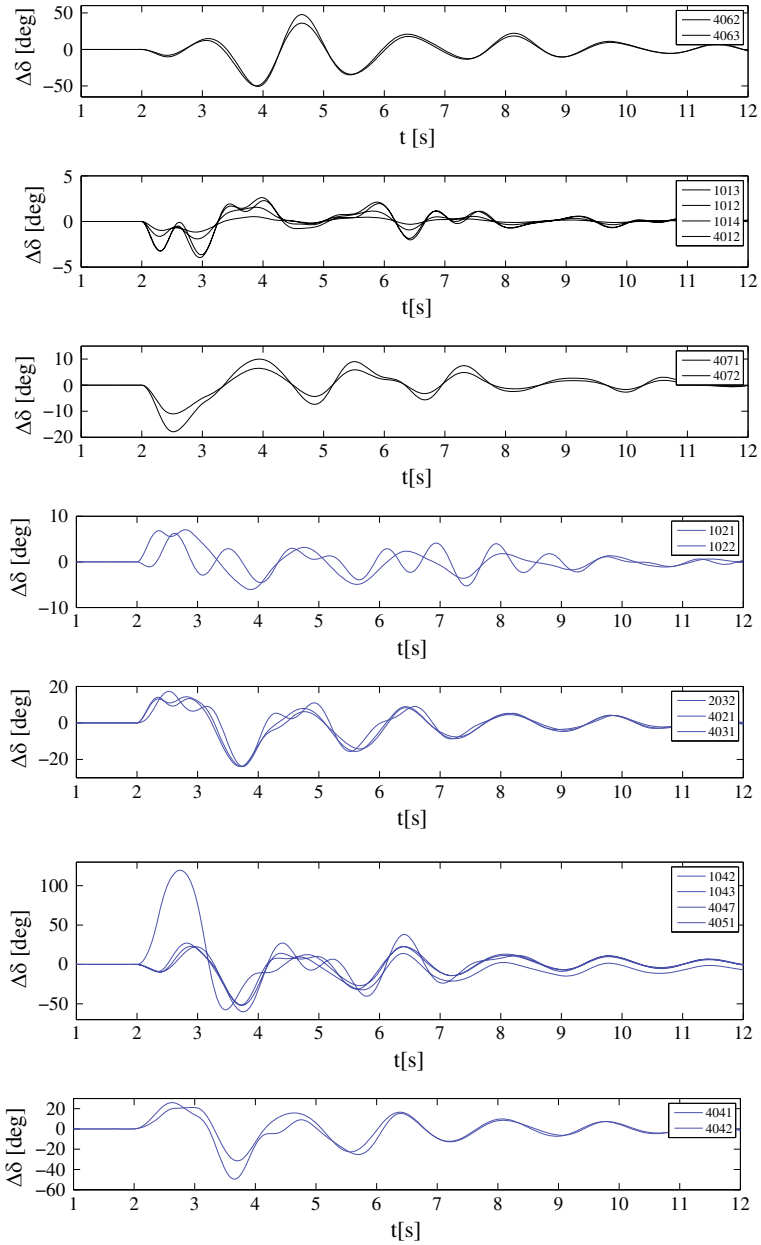
Fig. 9.6 Coherent sub-groups identified after the fault at bus B_{4047}

Table 9.4 Largest Koopman modes for the different faults

Fault	Mode j	Growth rate	Frequency (Hz)	Norm $\|\tilde{v_j}\|$
4031	3	0.99797	0.616	60.08
1021	2	0.99108	0.569	8.539
4063	8	0.99748	0.595	73.43
1012	3	0.99798	0.613	78.68

ing KMs and their frequencies according to the rest of the faults. Figure 9.7a–d show the distribution of the largest Koopman Modes.

Same groups (CG_1 and CG_2) and subgroups are obtained, nevertheless some slightly differences can be observed. For the fault located in bus B_{4063}, it can be seen in Figure 9.7c that G_{4062} and G_{4063} are more related to the central group of generators than other cases. Sub-group S_7 can be joined with sub-group S_6. The other generators remained to the same sub-groups.

9.5 Test Cases: Gradual Increasing of Non-synchronous Generation

The impact of the integration of non-synchronous generation on the test system coherency is analysed by replacing some of the synchronous generators with back-to-back FRC with the same active and reactive power outputs in order to guarantee the same initial conditions. Note that the power outputs are fixed through the simulation. Two scenarios are tested: first one, replaces synchronous generators in the Central-South area while the second one replaces in the North-External area.

9.5.1 Scenario Central-South

Case 1 (C_1) to Case 5 (C_5) represent the replacement of synchronous generation by the integration of non-synchronous generation based FRC gradually, in order to analyse different levels of power penetration. For example, C_1 considers the replacement of one generation only, and C_2 considers the replacement of two generators including the one in C_1, and so on. It is assumed that the dispersed generation is connected to one established substation. These five scenarios are summarised in Table 9.5:

Initially, KM are evaluated for fault at B_{4047} for cases presented above. Table 9.6 shows the KM variation according to the replacement by non-synchronous in the system. It can be seen that the largest mode frequency stays in the same range. The same behaviour is obtained in the rest of the cases; however, the frequency tables variation are not presented here.
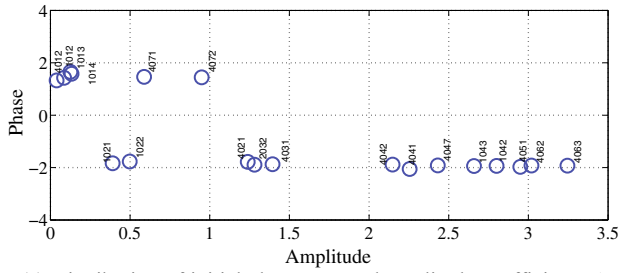
(a) Distribution of initial phases α_{ji} and amplitude coefficients A_{ji}
(after fault at bus B_{1021})
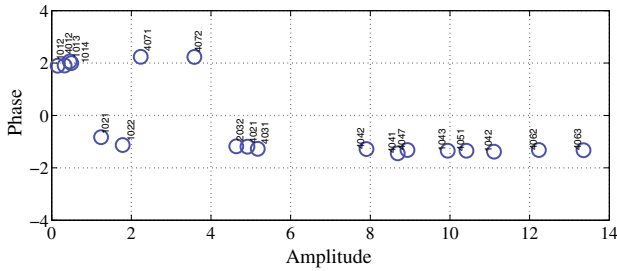
(b) Distribution of initial phases α_{ji} and amplitude coefficients A_{ji}
(after fault at bus B_{4031})
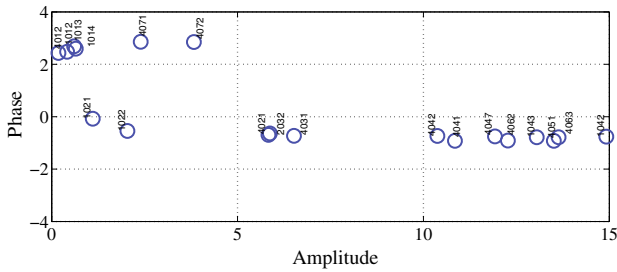
(c) Distribution of initial phases α_{ji} and amplitude coefficients A_{ji}
(after fault at bus B_{4063})
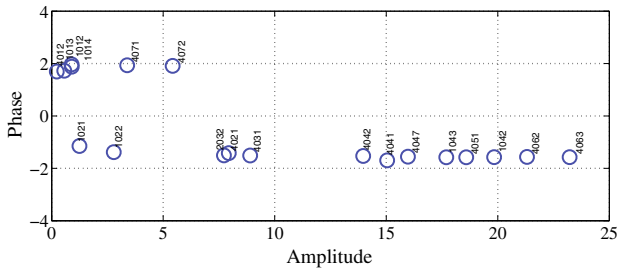
(d) Distribution of initial phases α_{ji} and amplitude coefficients A_{ji}
(after fault at bus B_{1012})

Fig. 9.7 Distribution of initial phases and amplitude coefficients for different faults

Table 9.5 Generator replacement for each case

Case	C_1	C_2	C_3	C_4	C_5
Generator	G_7	G_{16}	G_{17}	G_{14}	G_6
Bus	B_{1043}	B_{4051}	B_{4062}	B_{4042}	B_{1042}
Power (MW)%	1.2	5.4	9	13	15

Table 9.6 Koopman largest mode variation (fault 4047)

Mode j	Growth rate	Frequency (Hz)	Norm $\|\tilde{v}_j\|$	Case
8	0.99748	0.5957	73.43	BC
15	0.99721	0.5908	75.54	C_1
3	0.99894	0.5714	52.31	C_2
6	0.99813	0.5661	19.23	C_3
10	0.99789	0.5986	22.17	C_4
4	0.99978	0.6542	9.365	C_5

KM for C_2 is plotted in Figure 9.8. The replacement by non-synchronous gener-
ation in the first two cases does not affect the coherent groups/sub-groups analysed
before. Figure 9.8 shows the KM for C_3 and the significant changes for cases C_4
and C_5 with the circle points variation of the generators concerned. After the third
case, with the replacement of generator G_{4062}, generator G_{4063} (both in the South
area) become to swing and be more coherent with generators in the Central area.
Generators in the *North* and *External* area keep swinging together.

For the fault in bus B_{4031}, first two cases of non-synchronous generation replace-
ment do not show change in the coherent sub-groups as the former analysis. From the
third case, significant changes can be observed in the KM sub-groups. The sub-groups
based on the amplitude coefficient of the respective mode show that the Central area
have become more separated. Specially in case 4, where generator G_{1042} moves
closer to G_{4063} (Figure 9.9).

Generator G_{1042} is becoming more separate in C_4 for post-faults in buses B_{4063},
B_{4012} and B_{4021} as it can be seen in Figures 9.10 and 9.11 (Figure 9.12).

9.5.2 Scenario North-External Area

This scenario considers the replacement of some generators in the North and External
area. The location of these new cases in the system are presented in Table 9.7. It is
applied the same gradual replacement from the previous scenario.
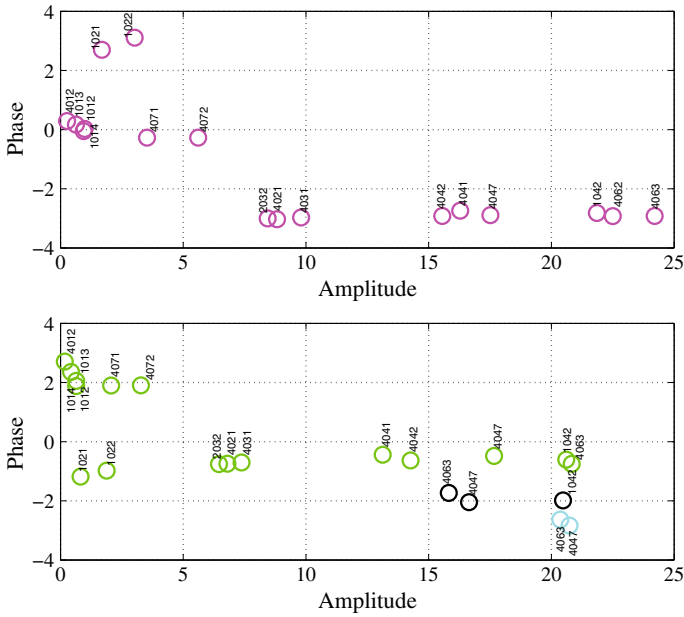
Fig. 9.8 Coherency identification including non-synchronous generation (after fault at bus B_{4047})
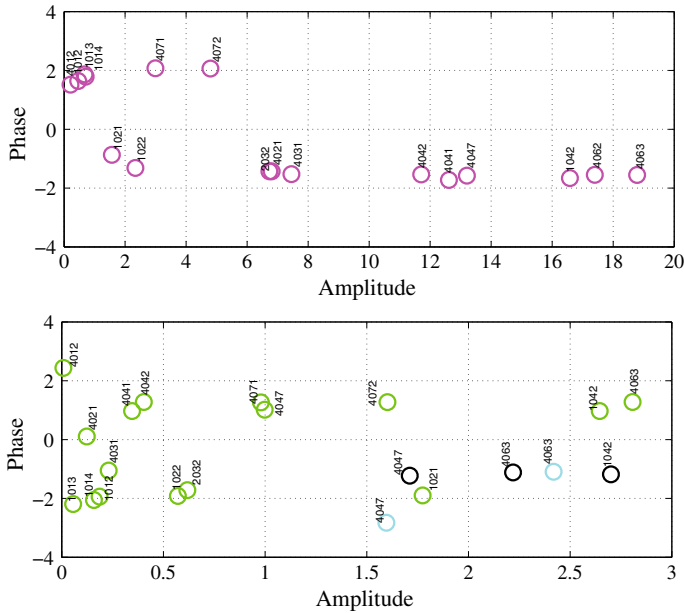
Fig. 9.9 Coherency identification including non-synchronous generation (after fault at bus B_{4031})
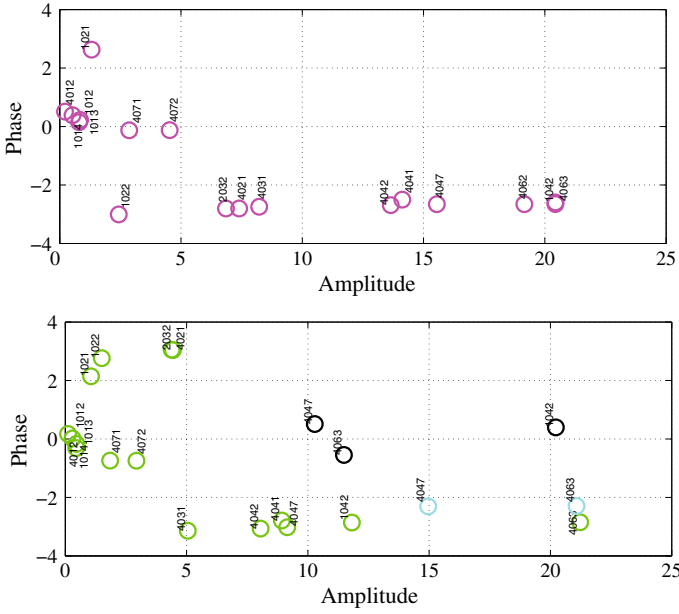
Fig. 9.10 Coherency identification including non-synchronous generation (after fault at bus B_{4063})
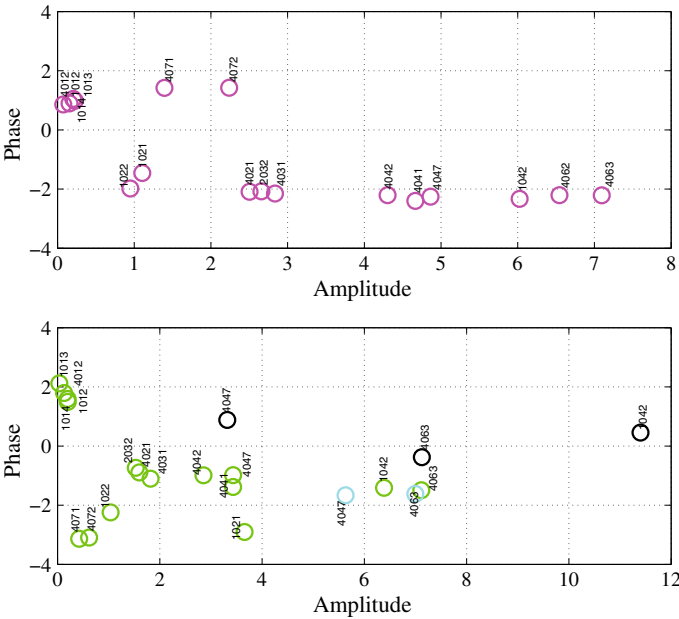
Fig. 9.11 Coherency identification including non-synchronous generation (after fault at bus B_{1021})
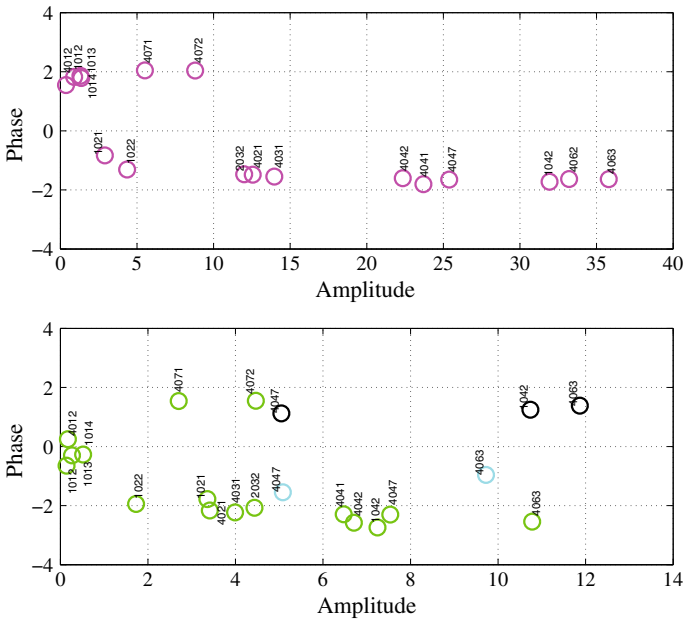
Fig. 9.12 Coherency identification including non-synchronous generation (after fault at bus B_{1012})

Table 9.7 Generator replacement for each case (North-Ext)

Case	C_1	C_2	C_3
Generator	G_2	G_{12}	G_{19}
Bus	B_{1013}	B_{1012}	B_{4071}
Power (MW)%	3	6	9

The replacement by non-synchronous generation in the first two cases remains the sub-groups previously identified for the same faults located. After the replacement of the third generator, for the fault located in bus B_{4031}, makes the generator G_{1021} be more coherent with the sub-group of generators G_{4012} and G_{1014}, whereas for the faults located in buses B_{1021} and B_{1012} the generator G_{1021} be more coherent with the generators G_{2032}, G_{4031} and G_{4021}. This is shown in Figures 9.13, 9.14 and 9.15 (Table 9.8).
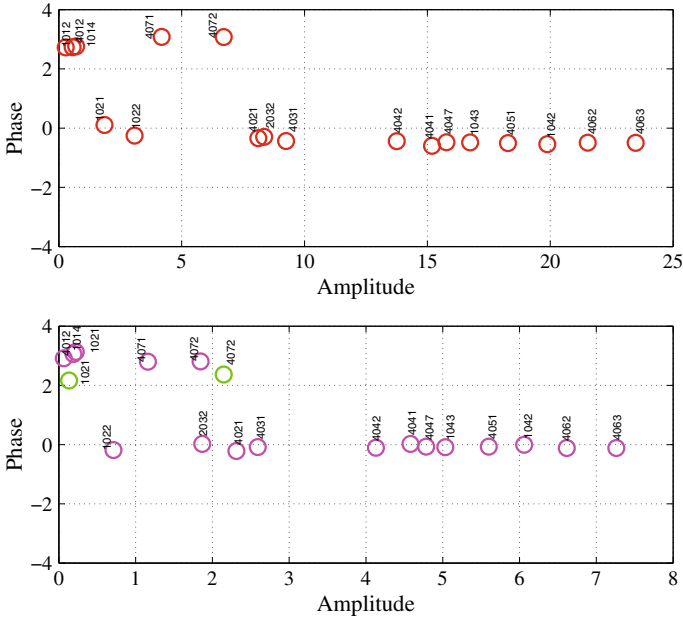
Fig. 9.13 Coherency identification including non-synchronous generation (after fault at bus B_{4031})
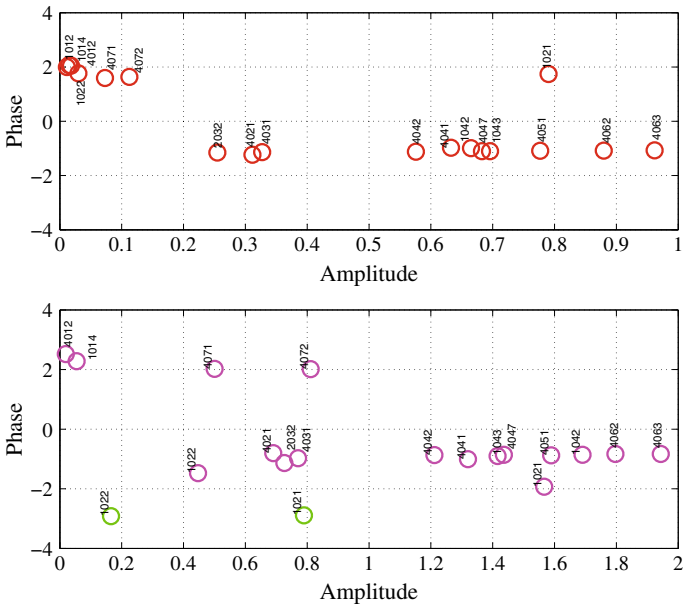
Fig. 9.14 Coherency identification including non-synchronous generation (after fault at bus B_{1021})
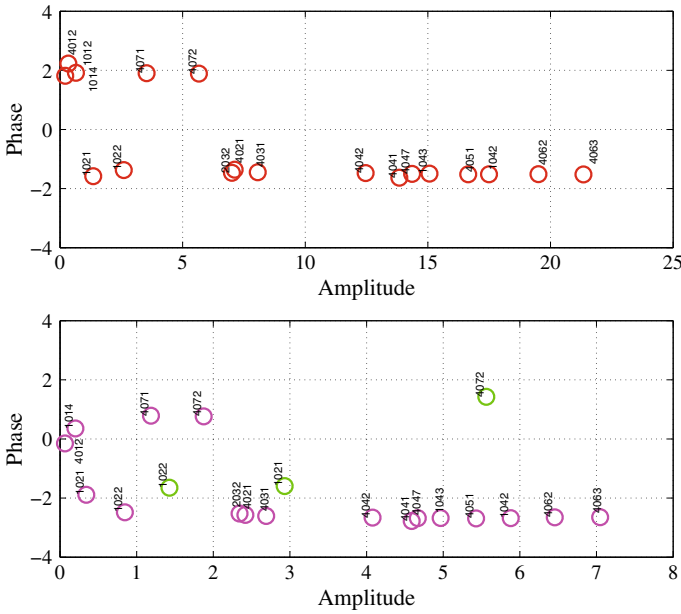
Fig. 9.15 Coherency identification including non-synchronous generation (after fault at bus B_{1012})

Table 9.8 Koopman largest mode variation (fault 4031)

Mode j	Growth rate	Frequency (Hz)	Norm $\lVert \tilde{v}_j \rVert$	Case
2	0.99797	0.6165	60.08	BC
5	0.99769	0.6097	54.68	1
2	0.99868	0.6552	13.73	2
8	0.9979	0.6349	16.91	3

9.6 Conclusions and Future Work

The document presents an original contribution to the application of the KMA for the coherency pattern identification of power systems system with a gradual large inclusion of non-synchronous generation.

The application of KMA to the studied power system (Nordic 32 system) showed the variation of the coherent groups through the gradual inclusion of power converters. The results show that if a minor replacement of synchronous machines the coherency is not affected, but after the replacement of the 13% of the power in the system, some significant changes in the coherent groups can be identified.

The coherent groups in one operative area can be altered by the increasing inclusion of non-synchronous generation becoming either groups apart or be more coherent with other groups. A general comment can be made in relation to the fault location

since the coherency phenomena is related to it. After studying several faults incorporating the gradual increasing of full converters. The effect of a fault close to a specific generator in the form of acceleration is propagated, and reflected around the closer generators in the form of synchronizing power. Therefore, with the replacement of synchronous machines (inertia reduction), the coherent groups separate different, as the amplitude coefficient separation in the KM showed.

The applied coherency identification method provides a direct calculation and a powerful graphical visualisation tool for observing the coherent groups/subgroups according to the measurements obtained while the dynamics are changing.

This tool has the potential to be integrated in real-time simulator systems and be improved by a sliding window, which means can be applied in situational awareness alert system or control loops design involved in FRC.

KMA application is concise with the slow coherency theory, however with the reduction of inertia in the system, the fault location change the coherent groups in the system.

Future work requires the addition of PSS to non-synchronous generation and other control loops, e.g. synthetic inertia in order to diminish the oscillations.

References

1. Bjornstedt, J.: Integration of non-synchronous generation—frequency dynamics. PhD thesis (2012)
2. Chamorro, H.R., Ordonez, C.A., Peng, J.C., Ghandhari, M.: Non-synchronous generation impact on power systems coherency. IET Gener. Transm. Distrib. **10**(10), 2443–2453 (2016)
3. Bongiorno, M., Petterson, A.: Development of a method for evaluation of wind turbines ability to fulful Swedish grid codes. Tech. Rep. Elforsk Rapport **09**(25) (2009)
4. Blaabjerg, F., Liserre, M., Ma, K.: Power electronics converters for wind turbine systems. IEEE Trans. Ind. Appl. **48**(2), 708–719 (2012)
5. Hui, J., Jain, P.: Power management and control of a wind energy conversion system (WECS) with a fuzzy logic based maximum power point tracking (MPPT). In: IECON 2012—38th Annual Conference on IEEE Industrial Electronics Society, pp. 5966–5971 (2012)
6. Lalor, G., Mullane, A., OMalley, M.: Frequency control and wind turbine technologies. IEEE Trans. Power Syst. **20**(4), 1905–1913 (2005)
7. Wang, Y., Delille, G., Bayem, H., Guillaud, X., Francois, B.: High wind power penetration in isolated power systems—assessment of wind inertial and primary frequency responses. IEEE Trans. Power Syst. **28**(3), 2412–2420 (2013)
8. Chamorro, H.R., Sanchez, A.C., Overjordet, A., Jimenez, F., Gonzalez-Longatt, F., Sood, V.K.: Distributed synthetic inertia control in power systems. In: 2017 International Conference on Energy and Environment (CIEM), pp. 74–78 (2017)
9. Muljadi, E., Gevorgian, V., Singh, M., Santos, S.: Understanding inertial and frequency response of wind power plants. In: IEEE Power Electronics and Machines in Wind Applications (PEMWA), pp. 1–8 (2012)
10. Brisebois, J., Aubut, N.: Wind farm inertia emulation to fulfill hydro-Quebec specific need. In: IEEE Power and Energy Society General Meeting, pp. 1–7 (2011)

11. Chavez, H., Baldick, R., Sharma, S.: Regulation adequacy analysis under high wind penetration scenarios in ERCOT nodal. IEEE Trans. Sustain. Energy **3**(4), 743–750 (2012)
12. Finley, A., Kosterev, D.: Planning efforts to evaluate dynamic response of increased penetration of variable generation within the western interconnection. In: IEEE Power and Energy Society General Meeting, pp. 1–8 (2012)
13. Gautam, D., Vittal, V., Harbour, T.: Impact of increased penetration of DFIG-based wind turbine generators on transient and small signal stability of power systems. IEEE Trans. Power Syst. **24**(3), 1426–1434 (2009)
14. Chamorro, H.R., Ghandhari, M., Eriksson, R.: Influence of the increasing non-synchronous generation on small signal stability. In: IEEE PES General Meeting|Conference & Exposition, National Harbor, pp. 1–5 (2014)
15. Eftekharnejad, S., Vittal, V., Heydt, G.T., Keel, B., Loehr, J.: Small signal stability assessment of power systems with increased penetration of photovoltaic generation: a case study. IEEE Trans. Sustain. Energy **4**(4), 960–967 (2013)
16. Bueno, P.G., Hernández, J.C., Ruiz-Rodriguez, F.J.: Stability assessment for transmission systems with large utility-scale photovoltaic units. IET Renew. Power Gener. **10**(5), 584–597 (2016)
17. Naik, P., Qureshi, W., Nair, N.-K.: Identification of coherent generator groups in power system networks with wind-farms. In: Universities Power Engineering Conference (AUPEC) pp. 1–5 (2011)
18. Gallai, A., Thomas, R.: Coherency identification for large electric power . IEEE Trans. Circuits Syst. **29**(11), 777–782 (1982)
19. Lee, S.T.Y., Schweppe, F.C.: Distance measures and coherency recognition for transient stability equivalents. IEEE Trans. Power Appar. Syst. **92**(5), 1550–1557 (1973)
20. Podmore, R.: Identification of coherent generators for dynamic equivalents. IEEE Trans. Power Appar. Syst. **97**(4), 1344–1354 (1978)
21. Jonsson, M., Begovic, M., Daalder, J.: A new method suitable for real-time generator coherency determination. IEEE Trans. Power Syst. **19**(3), 1473–1482 (2004)
22. Chow, J., Galarza, R., Accari, P., Price, W.: Inertial and slow coherency aggregation algorithms for power system dynamic model reduction. IEEE Trans. Power Syst. **10**(2), 680–685 (1995)
23. Winkelman, J.R., Chow, J., Bowler, B.C., Avramovic, B., Kokotovic, P.: An analysis of interarea dynamics of multimachine systems. IEEE Trans. Power Appar. Syst. **PAS-100**(2), 754–763 (1981)
24. Agrawal, R., Thukaram, D.: Support vector clustering-based direct coherency identification of generators in a multi-machine power system. IET Gener. Transm. Distrib. **7**(12), 1357–1366 (2013)
25. Djukanovic, M., Sobajic, D.J., Pao, Y.H.: Artificial neural network based identification of dynamic equivalents. Electric Power Syst. Res. **24**(1), 39–48 (1992)
26. Lino, O., Fette, M., Dong, Z., Ramirez, J.: Nonlinear approaches for dynamic equivalencing in power systems. In: Power Systems Conference and Exposition, pp. 1306–1314 (2006)
27. Wang, M.-H., Chang, H.-C.: Novel clustering method for coherency identification using an artificial neural network. IEEE Trans. Power Syst. **9**(4), 2056–2062 (1994)
28. Verma, K., Niazi, K.R.: Generator coherency determination in a smart grid using artificial neural network. In: IEEE Power and Energy Society General Meeting, pp. 1–7 (2012)
29. Tianqi, L., Jun, W., Xuan, L., Xingyuan, L.: A fuzzy clustering method for coherent generator groups identification based on A-K. In: International Conference on Sustainable Power Generation and Supply, 2009. SUPERGEN 09, pp. 1–4, Apr 2009
30. Wang, S.-C., Lee, S.-C., Wu, C.-J.: Analysis of Taiwan power system dynamic performance and coherency identification of synchronous generators using fuzzy c-means clustering. In: Proceedings of SICE Annual Conference (SICE), pp. 1420–1425 (2011)
31. Gil, M., Rios, M., Gomez, O.: Coherency identification based on maximum spanning tree partitioning. In: IEEE PowerTech (POWERTECH), pp. 1–6 (2013)
32. Wei, J., Kundur, D.: A multi-flock approach to rapid dynamic generator coherency identification. In: IEEE Power and Energy Society General Meeting (PES), pp. 1–5 (2013)

33. Avdakovic, S., Becirovic, E., Nuhanovic, A., Kusljugic, M.: Generator coherency using the wavelet phase difference approach. IEEE Trans. Power Syst. **29**(1), 271–278 (2014)
34. Senroy, N.: Generator coherency using the Hilbert Huang transform. IEEE Trans. Power Syst. **23**(4), 1701–1708 (2008)
35. Xu, G., Vittal, V.: Slow coherency based cutset determination algorithm for large power systems. IEEE Trans. Power Syst. **25**(2), 877–884 (2010)
36. Rios, M., Gomez, O.: Identification of coherent groups and PMU placement for inter-area monitoring based on graph theory. In: IEEE PES Conference on Innovative Smart Grid Technologies (ISGT Latin America) pp. 1–7 (2011)
37. Susuki, Y., Mezic, I.: Nonlinear Koopman modes and coherency identification of coupled swing dynamics. IEEE Trans. Power Syst. **26**(4), 1894–1904 (2011)
38. Petersen, K.E.: Ergodic Theory. English, Reprint edition. Cambridge University Press, Cambridge (1989)
39. Lasota, A., Mackey, M.C.: Chaos.: Fractals, and Noise Stochastic Aspects of Dynamics. English, 2nd edn. Springer, New York (1998)
40. Raak, F., Susuki, Y., Hikihara, T., Chamorro, H.R., Ghandhari, M.: Partitioning power grids via nonlinear Koopman mode analysis. In: Innovative Smart Grid Technologies Conference (ISGT), pp. 1–5 (2014)
41. Mezic, I.: Spectral properties of dynamical systems, model reduction and decomposition. Nonlinear Dyn. **41**(1–3), 309–325 (2005)
42. Susuki, Y., Mezic, I.: Nonlinear Koopman modes of coupled swing dynamics and coherency identification. In: IEEE Power and Energy Society General Meeting, pp. 1–8 (2010)
43. Susuki, Y., Mezic, I., Raak, F., Hikihara, T.: Applied Koopman operator for power systems technology. Nonlinear Theory Appl. **7**(4), 430–459 (2016)
44. Bialek, J.: Why has it happened again? Comparison between the UCTE blackout in 2006 and the blackouts of 2003. In: IEEE Power Tech Lausanne, pp. 51–56 (2007)
45. Alsafih, H.A., Dunn, R.: Determination of coherent clusters in a multi-machine power system based on wide-area signal measurements. In: IEEE Power and Energy Society General Meeting, pp. 1–8 (2010)
46. Koch, S., Chatzivasileiadis, S., Vrakopoulou, M., Andersson, G.: Mitigation of cascading failures by real-time controlled islanding and graceful load shedding. In: Bulk Power System Dynamics and Control (iREP)-VI
47. Terzija, V.: Wide area monitoring protection and control—WAMPAC. In: IET-UK International Conference on Information and Communication Technology in Electrical Sciences (ICTES 2007), Dec 2007
48. Moreno, R., Rios, M., Torres, A.: Security schemes of power systems against blackouts. In: 2010 iREP Symposium on Bulk Power System Dynamics and Control (iREP)-VIII (iREP), pp. 1–6, Aug 2010
49. Larsson, S., Ek, E.: The black-out in southern Sweden and eastern Denmark, September 23, 2003. In: IEEE Power Engineering Society General Meeting, vol. 2, pp. 1668–1672, June 2004
50. Norris, S., Shao, H., Bialek, J.: Considering voltage stress in preventive islanding. In: PowerTech (POWERTECH), 2013 IEEE Grenoble, Grenoble, pp. 1–6 (2013)
51. You, H., Vittal, V., Yang, Z.: Self-healing in power systems: an approach using islanding and rate of frequency decline based load shedding. IEEE Trans. Power Syst. **18**(1), 174–181 (2003)
52. Hiyama, T.: Coherency-based identification of optimum site for stabilizer applications. IEE Proc. C (Gener. Transm. Distrib.) **130**(2), 71–74 (1983)
53. Parsa, M., Toyoda, J.: Slow-coherency based composite mode oscillatory stabilization by means of a hybrid PSS. IEEE Trans. Power Syst. **4**(4), 1499–1506 (1989)
54. Zarghami, M., Crow, M., Jagannathan, S.: Nonlinear control of FACTS controllers for damping interarea oscillations in power. IEEE Trans. Power Deliv. **25**(4), 3113–3121 (2010)
55. Chenine, M., Ullberg, J., Nordstrom, L., Wu, Y., Ericsson, G.: A framework for wide-area monitoring and control systems interoperability and cybersecurity. IEEE Trans. Power Deliv. **29**(2), 633–641 (2014)

56. Sun, K., Luo, X., Wong, J.: Early warning of wide-area angular stability problems using synchrophasors. In: IEEE Power and Energy Society General Meeting, pp. 1–6, July 2012
57. Li, W., Bose, A.: A coherency based rescheduling method for dynamic security. IEEE Trans. Power Syst. **13**(3), 810–815 (1998)
58. Diao, R., Vittal, V., Logic, N.: Design of a real-time security assessment tool for situational awareness enhancement in modern power. IEEE Trans. Power Syst. **25**(2), 957–965 (2010)
59. Wang, S., Lu, S., Lin, G., Zhou, N.: Measurement-based coherency identification and aggregation for power systems. In: IEEE Power and Energy Society General Meeting, pp. 1–7, July 2012
60. Nath, R., Lamba, S., Rao, K.S.P.: Coherency based system decomposition into study and external areas using weak coupling. IEEE Trans. Power Appar. Syst. **PAS-104**(6), 1443–1449 (1985)
61. Yusof, S.B., Rogers, G., Alden, R.T.H.: Slow coherency based network partitioning including load. IEEE Trans. Power Syst. **8**(3), 1375–1382 (1993)
62. Tang, K., Venayagamoorthy, G.K.: Online coherency analysis of synchronous generators in a power system. In: 2014 IEEE PES Innovative Smart Grid Technologies Conference (ISGT), pp. 1–5, Feb 2014
63. Stadler, J., Renner, H., Kock, K.: An inter-area oscillation based approach for coherency identification in power systems. In: Power Systems Computation Conference (PSCC), Wroclaw, pp. 1–6 (2014)
64. Padhy, B.P., Srivastava, S.C., Verma, N.K.: A coherency-based approach for signal selection for wide area stabilizing control in power systems. IEEE Syst. J. **7**(4), 807–816 (2013)
65. Susuki, Y., Mezic, I.: Nonlinear Koopman modes and power system stability assessment without models. IEEE Trans. Power Syst. **29**(2), 899–907 (2014)
66. Tu, J.H., Rowley, C.W., Aram, E., Mittal, R.: Koopman spectral analysis of separated flow over a finite-thickness flat plate with elliptical leading edge. In: AIAA Paper 2011, vol. 2864 (2011)
67. Rowley, C.W., Mezic, I., Bagheri, S., Schlatter, P., Henningson, D.S.: Spectral analysis of nonlinear flows. J. Fluid Mech. **641**, 115 (2009)
68. Budiic, M., Mohr, R., Mezic, I.: Applied Koopmanism. Chaos Interdiscip. J. Nonlinear Sci. **22**(4), 047 510 (2012)
69. Force, C.T.: Long term dynamics phase II final report. CIGRE (1995)
70. Chamorro, H.R., Ghandhari, M., Eriksson, R.: Coherent groups identification under high penetration of non-synchronous generation. In: IEEE Power and Energy Society General Meeting (PESGM), pp. 1–5 (2016)

Chapter 10
Appliance Identification Through Nonintrusive Load Monitoring in Residences

Christos Gogos and George Georgiou

Abstract Residential energy consumption forms a major part of the total energy expenditure. Consumers, power utilities, grid operators, electric appliance manufacturers, government agencies, and others are greatly interested in curbing the energy consumption, expecting in return financial and environmental rewards. Better understanding of how energy is consumed in residences will be crucial in developing trustworthy Demand Side Management (DSM) systems. This work presents state-of-the-art approaches for disaggregating power consumption in residences through nonintrusive load monitoring. Also, it contributes a new dataset of detailed power consumption data that was captured in a residence that was specially set up. The results show that by analyzing overlapping power patterns that electrical appliances generate, and a resident-level energy meter of adequate granularity, appliance identification becomes possible.

10.1 Introduction

Great benefit lurks in capturing detailed energy consumption of electrical appliances on residences. Through measuring, inefficient appliances and thriftless usage patterns can be identified. Once this occurs, replacing certain appliances and deferring the use of others should have positive financial and environmental results. Furthermore, studies [7] support that residence occupants that are informed about detailed consumption tend to exhibit behavior against energy waste.

Measuring how much electrical energy each customer consumes has been an intrinsic part of the electricity generation, transmission, distribution, and usage system, collectively known as the grid, since its inception. However, the size and the

C. Gogos (✉) · G. Georgiou
Department of Informatics and Telecommunications, University of Ioannina,
Kostakioi Campus, 47100 Arta, Greece
e-mail: cgogos@uoi.gr

G. Georgiou
e-mail: ggeorg@teiep.gr

complexity of the grid has been steadily increased through the years and new challenges have arisen. A multitude of electrical appliances directly associated with people's quality of life depend on the reliability and good operation of the grid. The advent of technology allows smart metering of electricity consumption at various levels of granularity. Furthermore, communities can be formed and synchronize their electricity consumption so as to put less stress on the grid and reap in return financial and environmental benefits.

An ideal system should be able to inform privileged entities about the consumption of each individual electrical appliance in real time and with aggregated values for certain periods of interest. In theory, this can be achieved by measuring the consumption of each appliance. Indeed, some appliances provide embedded mechanisms that report their energy consumption, while energy meters can be put in front of appliances not equipped with such mechanisms. This approach is known as ILM (Intrusive Load Monitoring) and also as DS (Distributed Sensing) but has several shortcomings that renders it largely impractical. Some disadvantages of ILM are that no widespread standard of sensing electricity consumption on electrical appliances exists yet, its high complexity of configuration and its high cost. Furthermore, ILM, as its name indicates, is intrusive and it presupposes highly motivated customers willing to keep an installation of several metering devices in their homes. These are major concerns regarding the scaling of such systems to large numbers of customers.

A different approach is NILM (Non-Intrusive Load Monitoring) [8], which is also known as Single Point Sensing (SPS). In this case, a single meter is installed in the circuit breaker panel of each home. Through analyzing the energy consumption patterns, individual appliances can be under certain circumstances identified. Of course, this method is less accurate than ILM but has the potential of being massively adopted by customers and utilities alike.

National-level deployments of smart meters have taken place or are planned to be applied in several countries worldwide. For example, member states of the European Union have committed to rolling out close to 200 million electricity smart meters, which accounts for 72% of the European consumers, by 2020.[1] The prospect of using NILM techniques through those smart meters is tempting. Nevertheless, smart meters are designed in order to mainly facilitate the utilities and not the individual consumers. The frequency of measurements that they report is low and there are open issues that have to be addressed like security, privacy, viable financial modeling, efficient big data handling, and others.

Several approaches that aim at accurate estimation of individual appliance consumption from a central metering point have been proposed and are reviewed in Section 10.3. These approaches can be broadly categorized into supervised learning and unsupervised learning. Our approach for the problem belongs to the latter category and uses Convolutional Neural Networks and Bidirectional Long Short-Term Memories over a new dataset that we also provide as part of this work.

The rest of this paper is organized as follows. The next section presents the NILM problem. Section 10.3 presents state-of-the art approaches for load disaggregation. In

---

[1]http://ses.jrc.ec.europa.eu/smart-metering-deployment-european-union

Section 10.4, public datasets of interest for the problem are presented. Section 10.5 presents a 3-month case study of monitoring electricity consumption for a single residence. Finally, conclusions are presented in Section 10.6.

10.2   Nonintrusive Load Monitoring

NILM uses a residence-level meter and continuously analyzes the energy consumption characteristics of a single house. Machine learning algorithms are typically used in order to identify changes at the power signal that can be attributed to the operation of certain appliances. A survey of NILM approaches for energy disaggregation can be consulted in [33].

In order to disaggregate electric power, appliance-specific measurable characteristic features, or signatures, are employed, that reveal information concerning its operation and consumption patterns. Proposed signatures may refer to operation of a device in steady state, or to the shape, size, and duration of its transient behavior. Transient signatures require high sampling rates and more complex hardware. Signatures proposed include active and reactive power draw, root mean square (RMS) voltage and current, power factor, V-I trajectory, transient waves, harmonics, electromagnetic inference, and electric noise on the voltage due to the abrupt switching of electrical devices. Finally, features like time of day, appliance run times, temperature, and light sensing have been employed to provide supplementary information about appliance operation. These features are used in order to differentiate the various devices that populate the electric network. These devices can be classified in: "on/off" devices that switch between two operation states, finite state devices that include a finite number of operating states and devices with a continuous range of power draw (the most challenging category to be distinguished from the aggregated signal).

10.3   Load Disaggregation Approaches

The disaggregation algorithms proposed in the literature in order to identify the consumption profile of each individual appliance are categorized into supervised and unsupervised methodologies. This categorization is based on whether a priori information and labeled data (a diary of which appliance changed state and when) is required for training.

The classifiers used by supervised algorithms need labeled datasets, that include the features and signatures of individual appliances, in order to be trained. Appliance signatures can be labeled online, based on real-time event detection, or offline, where each appliance is monitored for a certain amount of time separately by a metering device. In order to avoid this cumbersome procedure to obtain individual appliance data, one of the following strategies can be pursued:

- During the training phase, appliances are switched on sequentially and thus recognized from the aggregated load measurement.
- Publicly available open datasets with signatures of several household devices are used for training, thus minimizing user intervention.

10.3.1 Supervised Disaggregation

Supervised disaggregation algorithms can be of the event-less optimization type, or of the event-based pattern recognition type that relies on event detection to identify the appliances.

Optimization-based supervised algorithms model NILM as an optimization problem. They try to break down an unknown composite load into a set of identifiable signatures of known appliances in a load signature database. Then they find the combination of appliances that minimizes the mismatch with the unknown load, in terms of the features of the appliances in the database and the respective features extracted for the unknown load from conventional electrical measurements. Integer Programming [12] and Genetic Algorithms [2] have been applied to solve the problem. Discerning appliances with overlapping load signatures and the presence of unknown loads in the aggregated signal are the main issues of optimization-based supervised algorithms.

Starting with the seminal work of Hart [8], that used a simple clustering-based technique of real and reactive power changes, pattern recognition methods use clustering and mapping of the state changes to a feature space, in order to identify itemized energy consumption in the aggregated signal. A challenging issue in clustering, is to specify automatically the number of different clusters to be used, which corresponds to the number of constituent devices in the aggregated load.

In [26], given the measured aggregated real power signal and a detected steady-state change, a naïve Bayes classifier is used in order to compute the most likely state of each device, assuming independent states between devices. The classifier is trained independently on each device and then the trained classifiers are used together to disaggregate a set of devices.

A Support Vector Machine (SVM) with a Gaussian kernel for the classification process for registered appliances and a one-class SVM for the detection process of unregistered appliances were used in [14]. SVMs with linear, polynomial, and Radial Basis Function (RBF) kernels, trained and tested using spike train data resulting from a Fourier harmonic analysis of the input current waveform, in the presence of multiple devices, were also developed for signature extraction and device identification in [30]. Moreover, Artificial Neural Network (ANN)-based approaches to nonintrusive harmonic source identification were used, including a multilayer perceptron (MLP) and an RBF neural network.

A k-Nearest Neighbor Rule (k-NNR) recognizer has been used in [31] in order to identify different types of appliances and their energizing and de-energizing operation

statuses, based on transient features. Finally, some forms of Hidden Markov Models (HMMs) have also been applied [23].

We summarize on Figure 10.1 the main features of the supervised disaggregation algorithms reported in the present section, which constitute a representative, but not exhaustive, list of the methodologies presently used in the literature.

10.3.2   Unsupervised Disaggregation

Unsupervised NILM methodologies are more adapted for real-time applications, since, unlike supervised NILM techniques, they do not require labeled datasets. By minimizing supplemental costs and human interaction, they may constitute a promising, inexpensive, large-scale, load disaggregation alternative. According to the data needed for energy disaggregation, the unsupervised approaches can be categorized into:

- Approaches that require unlabeled training submetered appliance data in order to build the appliance models and populate their appliance database (mostly HMM based techniques)
- Approaches that require prior submetered appliance data to be collected for training from known houses and then they can be further applied to unknown houses (mostly deep learning based techniques) and lastly
- Approaches that do not demand for submetered appliance data or prior knowledge, in order to perform energy disaggregation.

HMMs [27], Factorial HMMs (FHMMs) and their extensions, as Additive Factorial HMMs with a convex optimization formulation of approximate inference [18], Conditional FHMMs (that do not scale well with an increase in the number of target appliances), Factorial Hidden Semi-Markov Models, Conditional Factorial Hidden Semi-Markov Models [25] and Hierarchical Dirichlet Process Hidden Semi-Markov Models (HDP-HSMMs) [13] have been utilized in the literature for load disaggregation.

Usually, each single appliance is modeled as an HMM, i.e., a Markov statistical process with the finite set of its hidden (not measurable) discrete states representing the time series of the unobservable steady states of operation of the appliance. It is the output of the model that is visible to the observer. The Markov property states that the conditional probability distribution of a hidden state depends only on the value of the hidden state at the immediately previous time moment, that is, all the information in the sequence of states that preceded, is incorporated in that value. Then super-state HMMs or an FHMM, which can model the interaction of several independent processes, are used to model a household. In a super-state HMM, the output measurement corresponds to the measurement of a superposition of the states (the number of total states grows exponentially with the number of appliances). In a FHMM, the output at each time moment is the summation of measurements of the

SUPERVISED LEARNING

EVENTLESS OPTIMIZATION

EVENT BASED

INTEGER PROGRAMMING [12]

For appliances with multiple modes or of the same type operating simultaneously. A one-cycle waveform portion of every appliance is a prerequisite. No relearning needed when a new appliance is installed. Small computational burden. Cannot distinguish appliances with same waveforms. Wrong estimation for appliances with small current. Not applicable to cases where the operating modes cannot be defined. Accuracy rates of over 80% are reported.

HIERARCHICAL FHMM [23]

Discovers and clusters the signals of correlated devices. Hierarchical FHMMs work also with data of non independent devices and when even more than one devices change state at each time step. The relationship between devices is utilized to improve the speed and accuracy of inference.

CLUSTERING [8]

Uses measured average normalized power and RMS voltage as signatures. An edge detection algorithm finds the times and sizes of all step-like changes. Uses clustering of observed changes with determination of the appropriate number of clusters. Uses cluster pairing to construct on/off models. Suitable for on/off appliances. Not suitable for small, continuously variable or always 'on' appliances.

GENETIC ALGORITHMS [2]

Uses fuzzy clustering to find typical switch states of frequently occurring appliances, genetic algorithms to create Finite State Machines for different appliances and dynamic programming to find the best matching combination of switch events. Uses active power consumption, without a priori knowledge concerning special appliances. Detects recurring signatures of frequently occurring appliances' switching behavior and handles them as switch events (automatically detects patterns of Finite State Machines). Neglects rare and small events.

k-NNR RECOGNIZER [31]

After load energizing and deenergizing event detection is performed, a k-NNR recognizer based on transient features is used, to identify different types of appliances and their operation status under single-load and multiple load operation scenarios. In order to determine the class of an unknown feature vector, its nearest k neighbors, from each known classified vector in the training dataset, in terms of the squared Euclidean distance, are found, together with a majority voting mechanism . The Artificial Immune Algorithm with the Fischer criterion is employed to adaptively adjust the feature parameters and improve the recognizer identification performance, when a new type of appliance is added. Simplicity in computation and implementation. Overall recognition rates higher than 95% reported

NAÏVE BAYES [26]

Uses a naïve Bayes classifier to compute each device's most likely state, given a measured aggregate total and detected state step changes. Uses circuit-level energy measurements to be able to monitor small or variable power devices. Measures normalized real power and steady-state changes as an additional information source. Supposes non-correlated operation of devices. Can monitor devices with complex state or continuously variable power use. Training data is needed in order to detect each device. The classifier is trained independently on each device and the trained classifiers are used together to disaggregate a set of devices. Small resulting error rates are reported.

MLP, RBF NEURAL NETWORKS [30]

ANNS are used for classification, trained and tested using spike train data gathered from the Fourier analysis of the input current waveform in the presence of multiple devices. Several training algorithms (gradient descent with momentum, resilient backpropagation) were tested. All classifiers showed excellent classification results. Laboratory measurements and mathematically created data were used for training.

SVMs

Features are extracted from the electric current of each AC cycle by Principal Component Analysis (PCA). A SVM with a Gaussian kernel is used for the classification of registered appliances and a one-class SVM for detection of unregistered appliances. In the learning process electric current signals are used for all target appliances in advance. During recognition, features are extracted by the inner product of the appliance current signal and the PCA basis vectors. High accuracy is reported [14]. SVM-based models with linear, polynomial and RBF kernels are used to classify the presence or absence of combinations of several devices. Techniques are used to extend the two-class SVM classifier into a multiclass one [30].

Fig. 10.1 Supervised learning for NILM

individual chains and corresponds to the aggregate consumption of the household. An HMM is mainly characterized by the following quantities:

- The initial probability distribution of its states.
- A transition matrix that incorporates all the probabilities of transitioning from a certain state at a particular time moment to another state at the immediately successive time moment.
- An emission matrix representing the probabilities of observing a particular output measurement, given that the system is at a certain state in the same time moment.

Learning in an HMM consists of defining the above characteristics, given a sequence of measurement values of the observable output. Learning can be achieved by using, for example, expectation maximization [20], or iterative K-means [22]. Inferring in an HMM consists in retrieving the optimal sequence of hidden states, given its characteristics and a respective sequence of values of the observable output. Inferring (i.e., load disaggregation) can be achieved by using, for example, the sparse Viterbi algorithm [21], or a particle filter [6]. FHMMs are highly susceptible to local optima.

In [9, 32] a Graph Signal Processing (GSP)-based approach is proposed, by considering the load disaggregation problem as a single-channel blind source separation problem, on which to perform low-complexity classification of the acquired active power readings. Active power measurements are treated as a signal, indexed by the nodes of an undirected graph. The vertices of the graph correspond to the signal samples and the weights of the edges connecting the vertices reflect the degree of similarity between the nodes, enabling grouping of on/off events from the same appliance. Then, an optimization problem is defined and regularization on the constructed graph signal is applied, in order to find the signal with minimum variation.

Deep learning refers to machine learning approaches based on Artificial Neural Networks (ANNs) composed of many layers. By using this kind of architectures, it becomes possible to learn a hierarchy of features and these representations are invariant to local changes occurred in the input data. In general, deep learning models require big amounts of data, in order to generalize well. Deep learning architectures such as the biologically inspired Convolutional Neural Networks (CNNs) [3, 15], Recurrent Neural Networks (RNNs) [17], Long Short-Term Memories (LSTMs) [17], denoising Autoencoders (dAEs) that treat disaggregation as a denoising problem [17] and combinations of deep learning and HMMs [11] have been applied to the load disaggregation problem.

The main characteristics of the algorithms presented in this section for unsupervised disaggregation are reported in a condensed form on Figure 10.2.

10.4 Public Datasets

In 2011, the first publicly available data set for NILM research, the Reference Energy Disaggregation Dataset (REDD) was introduced [19]. REDD captured power data

UNSUPERVISED LEARNING

EM, K-MEANS, VITERBI, PF

In [20], a hierarchical hidden Markov model (HHMM) framework to model home appliances is proposed, for appliances with distinct power consumption profiles. A forward-backward algorithm, based on the framework of expectation maximization (EM), is formalized for the HHMM fitting process . In [22], an iterative K-means model fitting algorithm and a solver based on segment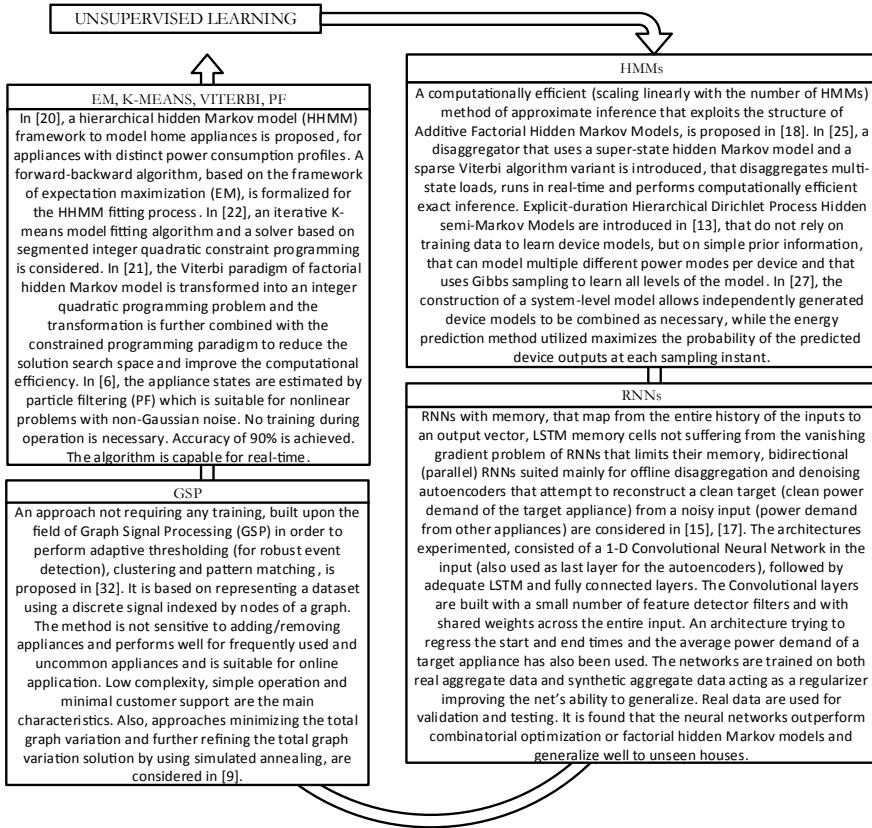ed integer quadratic constraint programming is considered. In [21], the Viterbi paradigm of factorial hidden Markov model is transformed into an integer quadratic programming problem and the transformation is further combined with the constrained programming paradigm to reduce the solution search space and improve the computational efficiency. In [6], the appliance states are estimated by particle filtering (PF) which is suitable for nonlinear problems with non-Gaussian noise. No training during operation is necessary. Accuracy of 90% is achieved. The algorithm is capable for real-time .

HMMs

A computationally efficient (scaling linearly with the number of HMMs) method of approximate inference that exploits the structure of Additive Factorial Hidden Markov Models, is proposed in [18]. In [25], a disaggregator that uses a super-state hidden Markov model and a sparse Viterbi algorithm variant is introduced, that disaggregates multi-state loads, runs in real-time and performs computationally efficient exact inference. Explicit-duration Hierarchical Dirichlet Process Hidden semi-Markov Models are introduced in [13], that do not rely on training data to learn device models, but on simple prior information, that can model multiple different power modes per device and that uses Gibbs sampling to learn all levels of the model. In [27], the construction of a system-level model allows independently generated device models to be combined as necessary, while the energy prediction method utilized maximizes the probability of the predicted device outputs at each sampling instant.

GSP

An approach not requiring any training, built upon the field of Graph Signal Processing (GSP) in order to perform adaptive thresholding (for robust event detection), clustering and pattern matching , is proposed in [32]. It is based on representing a dataset using a discrete signal indexed by nodes of a graph. The method is not sensitive to adding/removing appliances and performs well for frequently used and uncommon appliances and is suitable for online application. Low complexity, simple operation and minimal customer support are the main characteristics. Also, approaches minimizing the total graph variation and further refining the total graph variation solution by using simulated annealing, are considered in [9].

RNNs

RNNs with memory, that map from the entire history of the inputs to an output vector, LSTM memory cells not suffering from the vanishing gradient problem of RNNs that limits their memory, bidirectional (parallel) RNNs suited mainly for offline disaggregation and denoising autoencoders that attempt to reconstruct a clean target (clean power demand of the target appliance) from a noisy input (power demand from other appliances) are considered in [15], [17]. The architectures experimented, consisted of a 1-D Convolutional Neural Network in the input (also used as last layer for the autoencoders), followed by adequate LSTM and fully connected layers. The Convolutional layers are built with a small number of feature detector filters and with shared weights across the entire input. An architecture trying to regress the start and end times and the average power demand of a target appliance has also been used. The networks are trained on both real aggregate data and synthetic aggregate data acting as a regularizer improving the net's ability to generalize. Real data are used for validation and testing. It is found that the neural networks outperform combinatorial optimization or factorial hidden Markov models and generalize well to unseen houses.

Fig. 10.2 Unsupervised learning for NILM

from six residencies for a 19 days period. Since then, several researchers use it as a testbed for disaggregation algorithms. In 2012, the Building Level fully labeled dataset for Electricity Disaggregation (BLUED) was released [1] containing transitional power data from a single residence for an 8 days period. A number of other datasets focused on NILM research followed. A nonexclusive list of such datasets are Pecan Street (2013) [10], AMPds (2013) [24], iAWE (2013) [4], UK-DALE (2014) [16], and SustData (2014) [28].

Unfortunately, each dataset has its own characteristics like sample frequency, types of power data parameters measured, and others. These intricacies make comparison among disaggregation approaches difficult. In order to alleviate this problem NILMTK[2] toolkit was introduced [5]. NILMTK is an open-source project that enables parsing of several datasets, data preprocessing and statistical analysis. It

---

[2]http://nilmtk.github.io/

also includes a number of reference disaggregation algorithms and a set of accuracy metrics.

10.5   A Case Study

In this case study, power data of a single residence in Greece was monitored for a period spanning 117 days. During this period, occupants of the household kept conducting their usual activities. The metering frequency was at 1 reading per 10 s and in addition to the household aggregate, four appliances were also monitored: an electric oven, an electric water heater, a fridge, and a television. The physical quantities that were measured by each sensor were, the active power in Watts, the cumulative energy in Watt-hours and the voltage in Volts. The house used an oil heating system, while a solar water heater provided hot water during sunshine days.

It should be noted that participants in the experiment were informed about the data collected and gave consent about all possible uses of the data.

10.5.1   Experimental Setup

Three energy sensors were installed, one Meazon DinRail 3-Phase Advanced (Fig. 10.3) and two Meazon Izy Plugs (Fig. 10.4). Since DinRail is a three-phase sensor and the residence used single-phase power, one clamp was used for metering the mains while the other two were used for metering the electric water heater circuit and the electric oven circuit. All sensors sent readings wirelessly every 10 s to the gateway which was a Meazon Janus Box (Fig. 10.5). The gateway published the readings as MQTT topics that were captured from a daemon Python script running on a Raspberry PI and subsequently stored on an SQLite database creating a separate database file for each day. Sensors sent their readings to the gateway through the ZigBee protocol while the gateway and the Raspberry PI were connected to the same wired local network. Collected data were transformed to HDF5, which is the data format used by NILMTK. HDF5 embeds the data model metadata and supports effective storing and analyzing of data.

10.5.2   Descriptive Statistics

Data collection started at February 19, 2018 and ended at June 15, 2018. The percentage of the submetered energy was 58.76%. The remaining energy was consumed by devices like the washing machine, air conditioner, microwave oven, electric iron, and other electric appliances that were not submetered. The total energy consumed was 874.37 KWh. Dropout rates for the sensors were at 4.82% for the mains sensor

Fig. 10.3 Meazon DinRail
3-Phase advanced

Fig. 10.4 Meazon Izy Plug
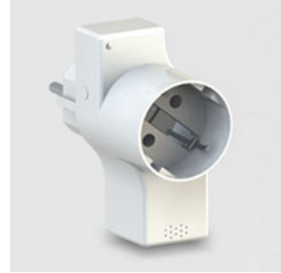(submetering sensor)

Fig. 10.5 Meazon Janus
Box (gateway)

(Fig. 10.6) and 4.04% and 0.12% for the fridge and television plug sensor, respectively. Dropout measures the proportion of missing samples which in our case was set to occur when a sensor failed to deliver a measurement in a period of 120 s. The television plug has the most favorite dropout value, since it was very close to the gateway, while the fridge plug was located further away and the main sensor even further away.

Figure 10.7 shows the percentage of consumption among devices that were submetered. Surprisingly, television was the device with the highest cumulative consumption during the experiment period. This device was an LCD 40 inches smart television (240 V, 40–60 Hz, 190 W). We found out that additionally to its typical use it was also used for background listening to radio stations, a fact that kept the device operating for long periods of time during the day. The second most consuming appliance among submetered ones was the fridge (220–240 V, 50 Hz, 230 W). Figure 10.8 shows the power consumption measured in a day.
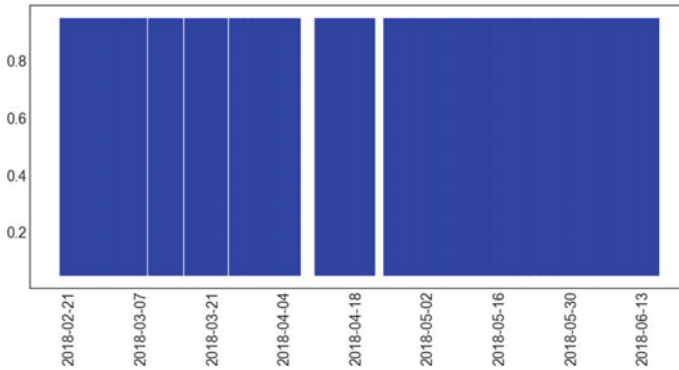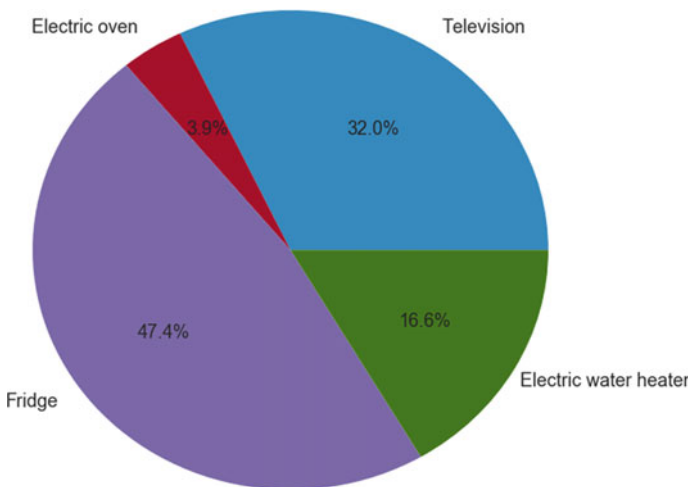
Fig. 10.6 Dropout rate for the mains sensor was at 4.82%

Fig. 10.7 Consumption among submetered electric appliances

10.5.3 Disaggregation Tests

We experimented with a multilayer architecture that uses Convolutional Neural Networks (CNNs) and Bidirectional LSTMs (Long Short-Term Memories).

A CNN consists of a number of convolutional and subsampling layers, followed by fully connected standard multilayer neural networks. Their architecture is designed to take advantage of the multidimensional structure of the input data and is achieved with local connections and tied weights and pooling, resulting in translation invariant features. A number of kernels (filters) is convolved with the input data to produce feature maps, which are then subsampled with pooling, over contiguous regions. An additive bias and a nonlinearity are applied to each feature map.
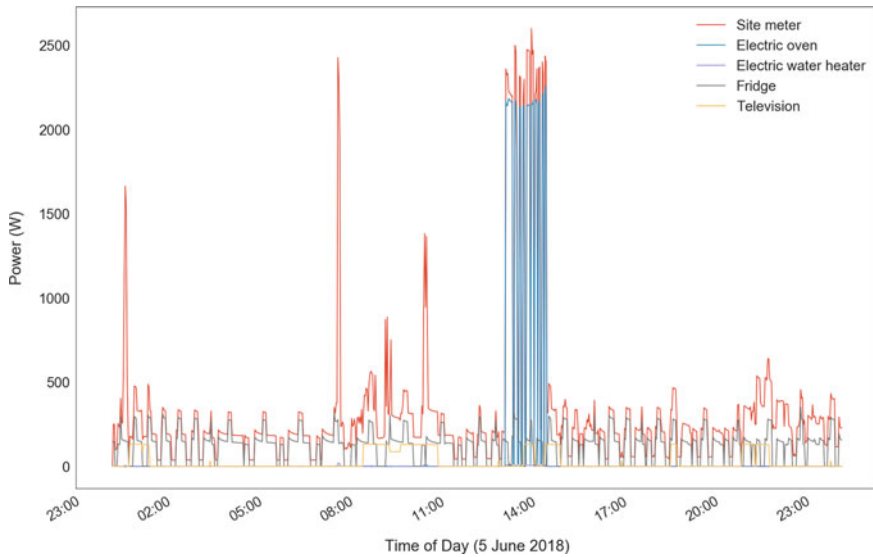
Fig. 10.8 Power data of a single day

LSTMs are a special kind of RNNs (Recurrent Neural Networks). In RNNs, feedback loops are allowed in each layer, from the output to the input, defining a recurrence relation over time steps that gives memory to the model. LSTMs have a different structure permitting to "shape" the information to throw away or store or output, through their forget gates, input gates and output gates, respectively. They can operate over sequences of vectors and can learn any causal time-varying mapping.

Bidirectional LSTMs make possible for future information to be reachable from the current state. Two opposite direction hidden layers, one accessing information in the forward direction and the other in the reverse one, are connected to the same output, so that the output layer takes information from past and future states. When all timestamps of the input sequence are available, one of the LSMTs is trained on the input sequence and the second one on its reverse copy.

The architecture used consisted of a 1D convolutional neural network in the input, having 16 output filters in the convolution, a stride length of 1, a kernel size (length of the 1D convolution window) of 4 and linear activation. It was followed by two stacked layers of bidirectional LSTMs having size 128 and 256, respectively. Then, a fully connected layer with 128 neurons and *tanh* type activation, fed a one neuron output layer with linear activation, in order to predict the disaggregated value.

The network was trained per target appliance. During the learning phase, in the forward pass, the network's output is initially computed for each specific network input (a window of aggregate power demand). Then the error between the computed network output and the target desired output (the power demand of the target appliance) is computed and used as the objective function. The connection weights and biases are then modified in the backward pass, in the direction that minimizes the

error. The neural networks learn through the modification of their weights. All the network weights were initialized randomly. Stochastic gradient descent was used to modify each weight and reduce the error. Because of a huge number of trainable parameters, large training datasets are required. Synthetic aggregate data were also used, which are easily obtainable by randomly combining data from single appliances over complete cycles of their operation. The use of synthetic data is important, since they act as a regularizer improving the network's capacity to generalize to unseen data. The multiple layers of neurons in the architectures used also improve learning a hierarchy of feature detectors from the data.

The network performance was evaluated by means of performance measures for binary classification, where the input is classified into one and only one, of two nonoverlapping classes. The use of *Precision*, *Recall*, *Accuracy*, and $F1$ scores in the evaluation of Machine Learning algorithms, is an example of borrowing from Information Retrieval and Information Extraction [29]. Given the number of True Positive (TP) events (positive events that are also classified as positive), True Negative (TN) events (negative events that are also classified as negative), False Positive (FP) events (negative events mistakenly classified as positive), and False Negative (FN) events (positive events mistakenly classified as negative), the total number of Positive (P) events, and the total number of Negative (N) events in ground-truth, we define

$$Accuracy = \frac{TP + TN}{P + N} \tag{10.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{10.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{10.3}$$

$$F1 = \frac{1}{\frac{\frac{1}{Precision} + \frac{1}{Recall}}{2}} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{10.4}$$

Accuracy measures the correctly predicted events in the total observations. *Precision* measures how many of the selected events are relevant (it is high when *FP* is low). *Recall* measures the percentage of the "right" events that were found (it is high when *FN* is low). *F1* is the harmonic mean of *Precision* and *Recall* (i.e., the reciprocal dual of arithmetic mean). The *F1* score gathers both *Precision* and *Recall* in one formula and takes an intermediate value between the *Precision* and *Recall* values.

Fridge Operation Disaggregation

After training the neural network with data collected during the time period from February 19, 2018 to April 30, 2018, we tested its learning with unseen data in the

Table 10.1 Metrics for the fridge disaggregation using RNN

Metric	Value (%)
Recall	97.17
Precision	89.99
Accuracy	89.86
*F*1 score	93.45

period after April 30, 2018. Results concerning a specific day are given in Figures 10.9, 10.10 and 10.11. Disaggregation results of the refrigerator and electric oven are reported.

A threshold is used so that, when an appliance's ground-truth or neural network estimated values transit above that threshold, indicates a transition from the "off" to the "on" state for that specific appliance, while an inverse transition indicates the passage from the "on" to the "off" state. These transitions were used in order to calculate the *Accuracy*, *Precision*, *Recall*, and *F*1 scores, which for the fridge disaggregation case are satisfactory and are reported on Table 10.1. The mean value of the absolute error between the estimated power and the ground-truth power amounts to 49.75 W, while the relative error in the total energy amounts to 18.68%.

In Figure 10.9, we report the ground-truth and estimated power for the refrigerator, during a specific day.
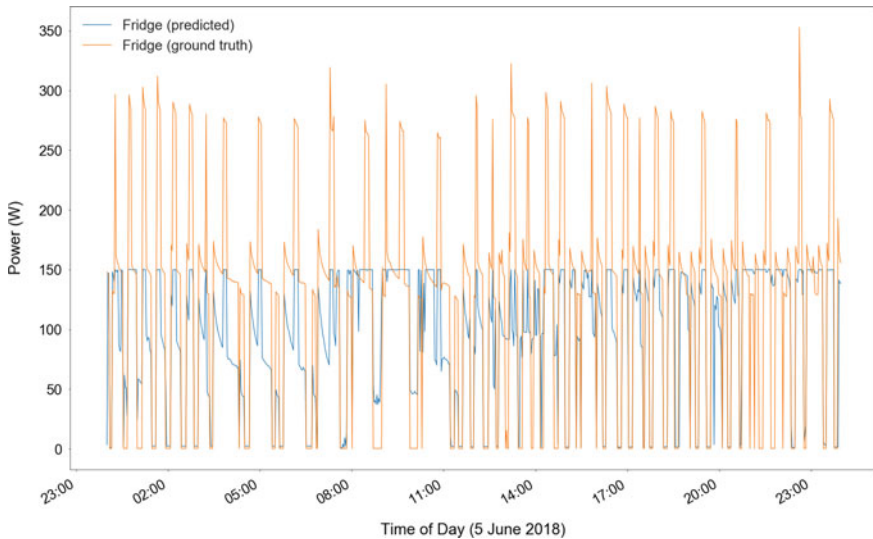

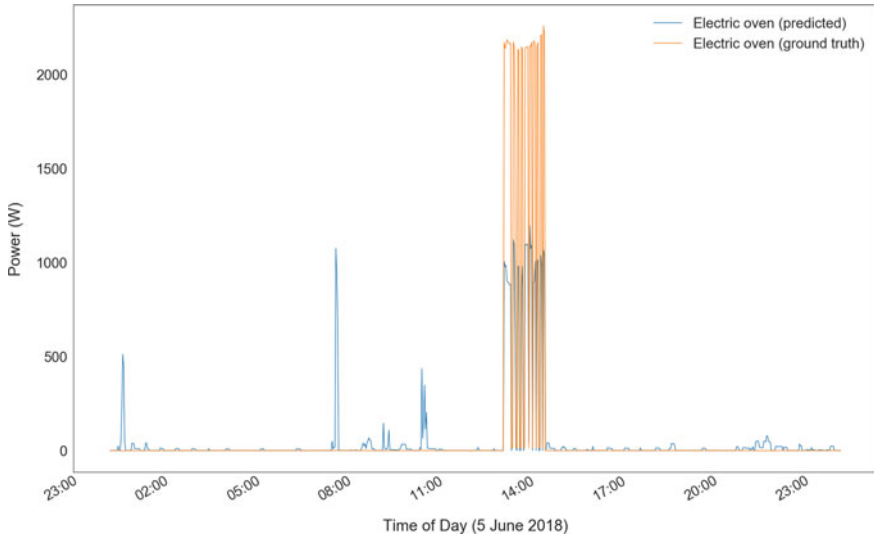**Fig. 10.9** Disaggregation of the fridge using RNN

Fig. 10.10   Disaggregation of the electric oven using RNN (predicted ground-truth)
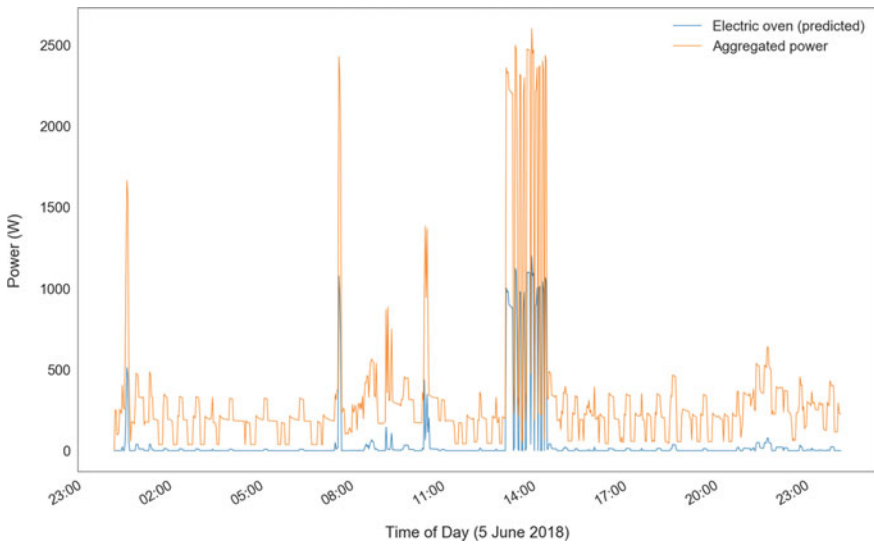

Fig. 10.11   Disaggregation of the electric oven using RNN (predicted-mains)

Table 10.2 Metrics for the electric oven disaggregation using RNN

Metric	Value (%)
Recall	44.61
Precision	24.58
Accuracy	73.65
*F*1 score	31.70

Electric Oven Operation Disaggregation

The respective results for the electric oven are given on Figure 10.10, while on Figure 10.11 we report the appliance's estimated power and the total aggregated power. The scores for the electric oven are reported on Table 10.2, while the mean value of the absolute error equals 56.03 W and the relative error in the total energy equals 28.10%.

Let us notice that the power demand of the electric oven (in the order of 2KW) is much higher than that of the fridge (which is in the range of 250W), so that these two appliances should be easily separated by the RNN. However, while the metrics of the fridge are satisfactory as one can conclude from Table 10.1, this is not true in the case of the electric oven. The low *Precision* score in Table 10.2 for the electric oven disaggregation can be justified by the presence of a high number of false positives (FP), as this becomes evident by inspection of Equation (10.2). Indeed, it becomes clear from Figure 10.10 that there exist predictions of positive events for the operation of the electric oven, which do not correspond to the electric oven ground-truth. From Figure 10.11, where the aggregated power demand is reported, it becomes evident that the false positive events correspond to peaks that exist on the aggregated power signal and which are due to another appliance in the electric network. These peaks are falsely perceived by the RNN as belonging to the electric oven appliance.

The dataset introduced in this paper can be found at https://github.com/chgogos/ NILMGR.

10.6 Conclusions

Much of the electricity is wasted. Every action that can optimize the way that we produce, distribute and consume electrical energy has the potential of big benefits for the planet at large. The stages that are responsible for the heavier losses are the generation stage and the final use stage of energy, while significantly smaller losses occur during the transmission and distribution stages. A great portion of electricity consumption is attributed to residences. It is believed that home owners don't have good intuition about the energy consumed by electrical appliances they use on a daily basis. NILM has the potential to improve this situation with minimal annoyance and involvement of the occupants.

NILM research started on the 80 s. Since 2011, NILM attracted much interest from the scientific community and the industry. Possible causes for this phenomenon might be the advent of smart meters, the machine learning flourishing and the need to curb electricity consumption for environmental and financial reasons. Several new approaches for appliance disaggregation were suggested and new ideas emerge at an accelerated pace. This paper presented a review of the state of the art for NILM, introduced a new dataset for the problem, and presented disaggregation results for specific appliances using RNNs.

Since our sensors support higher sampling frequencies, we plan to further investigate the problem by developing methods for the use of transient signatures of different appliances, which can be differentiated by the shape, or harmonic content of their transient behaviors. Moreover, it would be interesting to identify the effect that incorporated knowledge of external sources of information like weather, sunrise and sunset times, holidays and feast days, time of day, might have on the improvement of the accuracy of our predictions.

References

1. Anderson, K., Ocneanu, A., Benitez, D., Carlson, D., Rowe, A., Berges, M.: BLUED: a fully labeled public dataset for event-based non-intrusive load monitoring research. In: 2nd KDD Workshop on Data Mining Applications in Sustainability (SustKDD), pp. 12–16 (2012)
2. Baranski, M., Voss, J.: Genetic algorithm for pattern detection in NIALM Systems. In: Proceedings of IEEE International Conference on Systems Man and Cybernetics vol. 4, pp. 3462–3468 (2004)
3. Barsim, K.S., Mauch, L., Yang, B.: Neural network ensembles to real-time identification of plug-level appliance measurements. In: Proceedings of the 2016 NILM Workshop (2016)
4. Batra, N., Gulati, M., Singh, A., Srivastava, M.B.: It's different: insights into home energy consumption in India. In: Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings, pp. 1–8. ACM (2013)
5. Batra, N., Kelly, J., Parson, O., Dutta, H., Knottenbelt, W., Rogers, A., Singh, A., Srivastava, M.: NILMTK: An open source toolkit for non-intrusive load monitoring. In: Proceedings of the 5th International Conference on Future Energy Systems, pp. 265–276. ACM (2014)
6. Egarter, D., Bhuvana, V.P., Elmenreich, W.: PALDi: online load disaggregation via particle filtering. IEEE Trans. Instrum. Meas. **64**(2), 467–477 (2015)
7. Darby, S., et al.: The effectiveness of feedback on energy consumption. A Rev. DEFRA Lit. Metering, Billing Direct Disp. **486**(2006), 26 (2006)
8. Hart, G.W.: Nonintrusive appliance load monitoring. Proc. IEEE **80**(12), 1870–1891 (1992)
9. He, K., Stankovic, L., Liao, J., Stankovic, V.: Non-intrusive load disaggregation using graph signal processing. IEEE Trans. Smart Grid (2016)
10. Holcomb, C., Pecan street inc.: A test-bed for NILM. In: International Workshop on Non-Intrusive Load Monitoring, Pittsburgh, PA, USA (2012)
11. Huss, A.: Hybrid model approach to appliance load disaggregation: expressive appliance modelling by combining convolutional neural networks and hidden semi markov models (2015)

12. Inagaki, S., Egami, T., Suzuki, T., Nakamura, H., Ito, K.: Nonintrusive appliance load monitoring based on integer programming. Electr. Eng. Jpn. **174**(2), 18–25 (2011)
13. Johnson, M.J., Willsky, A.S.: Bayesian nonparametric hidden semi-markov models. J. Mach. Learn. Res. **14**(2), 673–701 (2013)
14. Kato, T., Cho, H.S., Lee, D., Toyomura, T., Yamazaki, T.: Appliance recognition from electric current signals for information-energy integrated network in home environments. In: International Conference on Smart Homes and Health Telematics, pp. 150–157. Springer, Berlin (2009)
15. Kelly, J., Knottenbelt, W.: Neural NILM: deep neural networks applied to energy disaggregation. In Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments, pp. 55–64. ACM (2015)
16. Kelly, J., Knottenbelt, W.: The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. Sci. Data **2**, 150007 (2015)
17. Kelly, D.: Disaggregation of domestic smart meter energy data. Ph.D. thesis, Imperial College London, Technology and Medicine (2016)
18. Kolter, J.Z., Jaakkola, T.: Approximate inference in additive factorial HMMs with application to energy disaggregation. In: Artificial Intelligence and Statistics, pp. 1472–1482, (2012)
19. Kolter, J.Z., Johnson, M.J.: REDD: a public data set for energy disaggregation research. In: Workshop on Data Mining Applications in Sustainability (SIGKDD), vol. 25, pp. 59–62. San Diego, CA (2011)
20. Kong, W., Dong, Z.Y., Hill, D.J.: A hierarchical hidden markov model framework for home appliance modelling. IEEE Trans. Smart Grid (2016)
21. Kong, W., Dong, Z.Y., Hill, D.J., Luo, F., Xu, Y.: Improving nonintrusive load monitoring efficiency via a hybrid programing method. IEEE Trans. Ind. Inform. **12**(6), 2148–2157 (2016)
22. Kong, W., Dong, Z.Y., Ma, J., Hill, D.J., Zhao, J. and Luo, F.: An extensible approach for non-intrusive load disaggregation with smart meter data. IEEE Trans. Smart Grid (2016)
23. Li, Y., Peng, Z., Huang, J., Zhang, Z., Son, J.H.: Energy disaggregation via hierarchical factorial HMM. In: Proceedings of the 2nd International Workshop on Non-Intrusive Load Monitoring (NILM) (2014)
24. Makonin, S., Popowich, F., Bartram, L., Gill, B., Bajic, I.V.: AMPds: a public dataset for load disaggregation and eco-feedback research. In: 2013 IEEE, Electrical Power and Energy Conference (EPEC), pp. 1–6. IEEE (2013)
25. Makonin, S.W.: Real-time embedded low-frequency load disaggregation. Ph.D. thesis, Simon Fraser University (2014)
26. Marchiori, A., Hakkarinen, D., Han, Q., Earle, L.: Circuit-level load monitoring for household energy management. IEEE Pervasive Comput. **10**(1), 40–48 (2011)
27. Mueller, J.A., Kimball, J.W.: Accurate energy use estimation for nonintrusive load monitoring in systems of known devices. IEEE Trans. Smart Grid (2016)
28. Pereira, L., Quintal, F., Gonçalves, R., Nunes, N.J.: SustData: a public dataset for ICT4S electric energy research. In ICT for Sustainability 2014 (ICT4S-14). Atlantis Press (2014)
29. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inf. Process. Manag. **45**(4), 427–437 (2009)
30. Srinivasan, D., Ng, W.S., Liew, A.C.: Neural-network-based signature recognition for harmonic source identification. IEEE Trans. Power Deliv. **21**(1), 398–405 (2006)
31. Tsai, M.-S., Lin, Y.-H.: Modern development of an adaptive non-intrusive appliance load monitoring system in electricity energy conservation. Appl. Energy **96**, 55–73 (2012)
32. Zhao, B., Stankovic, L., Stankovic, V.: On a training-less solution for non-intrusive appliance load monitoring using graph signal processing. IEEE Access **4**, 1784–1799 (2016)
33. Zoha, A., Gluhak, A., Imran, M., Rajasegarar, S.: Non-intrusive load monitoring approaches for disaggregated energy sensing: a survey. Sensors, **12**(12), 16838–16866 (2012)

Chapter 11
Management Suggestions for Process Control of Semiconductor Manufacturing: An Operations Research and Data Science Perspective

Marzieh Khakifirooz, Mahdi Fathi, Chen Fu Chien and Panos M. Pardalos

Abstract With advances in information and telecommunication technologies and data-enabled decision-making, smart manufacturing can be an essential component of sustainable development. In the era of the smart world, semiconductor industry is one of the few global industries that are in a growth mode to smartness, due to worldwide demand. The promising significant opportunities to reduce cost, boost productivity, and improve quality in wafer manufacturing is based on the integration or combination of simulated replicas of actual equipment, Cyber-Physical Systems (CPS) and regionalized or decentralized decision-making into a smart factory. However, this integration also presents the industry with novel unique challenges. The stream of the data from sensors, robots, and CPS can aid to make the manufacturing smart. Therefore, it would be an increased need for modeling, optimization, and simulation to the value delivery from manufacturing data. This paper aims to review the success story of smart manufacturing in semiconductor industry with the focus on data-enabled decision-making and optimization applications based on "Operations Research" (OR) and "Data Science" (DS) perspective. In addition, we will discuss future research directions and new challenges to this industry.

M. Khakifirooz (✉)
Tecnológico de Monterrey, Monterrey, Mexico
e-mail: mkhakifirooz@tec.mx

M. Fathi
Mississippi State University, Starkville, MS, USA
e-mail: fathi@ise.msstate.edu

C. F. Chien
National Tsing Hua University, Hsinchu, Taiwan
e-mail: cfchien@mx.nthu.edu.tw

P. M. Pardalos
University of Florida, Gainesville, FL, USA
e-mail: pardalos@ise.ufl.edu

11.1 Introduction

11.1.1 Industrial Revolution and "Industry 4.0"

The importance of national manufacturing strategies such as "Advanced Manufacturing Partnership" (AMP) of USA since 2011 [1], "Industry 4.0" of Germany since 2011 [2], "La Nouvelle France Industrielle" since 2013 [3], "Future of Manufacturing" of UK since 2013 [4], "Made in Sweden 2030" since 2013 [5], "Factories of the Future" of European Commission since 2014 [6], "Korea Manufacturing Innovation 3.0" since 2014 [7], "Industria Conectada 4.0" of Spain since 2014 [8], "Smart Industry" of Netherlands since 2014 [9], "Industry 4.1J" of Japan since 2015 [10], "Made in China 2025" since 2015 [11], "Fabbrica Intelligente" of Italy since 2015 [12], and "Innovation and Enterprise 2020 Plan" of Singapore since 2016 [13] have reemphasized the shifting standard of manufacturing and production system which led to the Fourth Industrial Revolution generation.

The industrial revolution stream drives the deployment of novel concepts for smart factories, new generation of monitoring and collaborating systems, or in general words the smart manufacturing system which it is built upon the CPS [14], "Internet of Things" (IoT) [15], and cloud and cognitive computing [16, 17]. The first step toward the smart manufacturing is the connectivity [18]. All the components in the industry must be connected to a single network, which is being allowed by the CPS and IoT which further confesses information interchange and alliance to attain a flexible and self-adaptive system of production. Moreover, by the integration of information and technologies, cloud and cognitive computing can facilitate the internet-based optimum interface and diagnostics, and can comprehend self-control system (self-learning, self-optimization, and self-awareness).

The fundamental concepts for designing smart manufacturing concerning the discipline and the precise distinction in their respective meaning and utilization are as follows [19]:

- Adaptation to human needs,
- Advance development of products and services,
- CPS,
- Corporate social responsibility,
- New systems in distribution and procurement,
- Self-organization,
- Smart factory.

The concepts mentioned above might experience several kinds of challenges and complications for smart manufacturing that may include technological, economical, social, political, and scientific issues [20]. This paper aims to review the area of science and technology challenges and point out the industry which is one of the most capital-intensive and complex that is semiconductor manufacturing industry.

11.1.2   *Semiconductor Industry and "Industry 4.0"*

In this era as a part of the technology roadmap for semiconductors driven by Moore's law [21] system scaling, there is more and more challenges by the poverty of resources and emergence of information technology. While the goal of semiconductor industry possesses the ability to continue technology migration to maintain overall performance, it is practically challenging to secure this objective due to the demand for appropriate action, together with all the steps required for the design to marketing. Therefore, the seamless interaction of smart manufacturing components such as big data, instant data, information technology (cloud, and multimode sensors), high-performance computing, mobile computing, and autonomous sensing and computing is necessary for driving "More Moore" (MM) technologies [22].

The "International Technology Roadmap for Semiconductors" (ITRS) [23] identified several critical limitations faced by semiconductor industry in the near future, will involve most, if not all, system integration, heterogeneous integration, heterogeneous components, external system connectivity, and factory integration.

ITRS determined a 35–40% less die cost [24], as one of the technical and reliability requirements to sustain MM technology. To achieve this goal, ITRS identified process integration, as one of the essential functional elements and critical challenges to stimulate the need for research and development and to meet a sustainable level of MM technology. The ITRS metrology chapter has underlined that the primary drivers in dealing with process integration are smart automotive, green energy, mobile communication systems, big data, and medical and health technologies [25].

Process integration, in particular, is dealing with technology and requirements associated with several phenomena such as:

- Cross leveraging factory integration technologies, across boundaries to achieve economies of scale.
- Attaining financial development goals while margins are decreasing.
- Increasing global restrictions on environmental issues.
- Dealing with the growing complexity.
- Achieving factory requirements such as capability, cost, equipment reliability, and productivity.
- Meeting adaptability, scalability and extensibility requirements of a profitable pioneering factory.
- Post-conventional Semiconductor manufacturing uncertainty (i.e., manufacturing requirements for new devices, timing uncertainty to identify new devices).
- Constantly responding to ever fluctuating, intricate business demands.

This paper is an extended version of [26] and aims to provide a systematic literature review on the scientific progress of the fourth industrial revolution ("Industry 4.0"—the most pointed national smart manufacturing strategy) with the perspective of OR&DS for semiconductor manufacturing. Most precisely, three research questions are given below

1. What would be the main challenges in the OR&DS point of view, enabling the industrial revolution in semiconductor manufacturing?
2. How are the OR&DS addressed the science and technology challenges in smart manufacturing?
3. What are the managerial suggestions from the integrated information of reviewed papers to prevail the unseen and future challenges in the path forward to the implementation of smart semiconductor manufacturing?

In Section 11.2, we identify the core challenges of wafer fabrication processes addressed in literature and the reviewing criteria are used for categorizing the findings and studies. In Section 11.3, we detail how these studies are considered the OR&DS fields into the intelligence semiconductor manufacturing, and how particular methods are distributed. Thereafter, from the gap in the literature, we propose some managerial suggestions in Section 11.4, for who are interested in walking into the field of semiconductor intelligence from the OR&DS perspective and in the domain of the "Industry 4.0". We conclude the paper in Section 11.5, by providing recommendations for further research and align our mindset for the next step.

11.2 Semiconductor Manufacturing Engineering

In semiconductor fabrication facilities (fabs), in order to fulfill the volatile demands of the high-mixed product, the related processes and electronic equipment are employed to produce Integrated Circuits (IC) with the help of a vast number of processing steps, batch processing models, sequence-dependent tool structures, the auxiliary resources [27] and recirculating flows. Therefore, this industry remains the most capital-intensive, for fully automated manufacturing systems [28]. The operations control of manufacturing facilities of semiconductor is known as tough task and is envisaged as one of the most composite manufacturing environments. One solution to deal with these difficulties is to choose the manufacturing and process data to analyze and modeling processes to empower factories in order to intensify an enhanced knowledge of the challenges associated with the production process and to grow visions which can develop prevailing procedures. Hereupon, this is very important to have enough understanding of the prevailing position of research about decision-making-based data engineering technologies in semiconductor industry and recognize fields for future research to maintain the further technologies for IC manufacturing. Therefore, this study aims to detect gaps in the existing works, develop significant research ideas, categorize existing research struggles and form a layout that will deliver different ideas related to the OR&DS area in smart IC manufacturing.

11.2.1 Challenges in Control of Semiconductor Manufacturing Process

Despite the sophisticated production process of wafer fabrication, the OR&DS techniques are using basically for the purpose of throughput enhancement and quality assurance. Regards the general application and intention of using OR&DS techniques, the main challenges in semiconductor manufacturing are categorized as follows:

- Photolithography process as the cutting-edge process and being bottleneck in the production process of semiconductor devices. The main challenge in the photolithography process is a misalignment between laser bean, wafer surface, and patterning mask, the error caused by this misalignment is called overlay error. Overlay error basically has a nonlinear relationship with overlay parameters and overlay parameters are not independent of each other.
- Large number of processing steps, batch tools, random equipment failures, re-entrant flows, sequence-dependent tool setups, and auxiliary resources for some process (i.e., photolithography process) are another source of challenges in semiconductor manufacturing process. Besides these facts, the semiconductor manufacturing equipment is extremely costly and to save the cost and time, the production schedule is mixed, or required to be patched. Dispatching the mixed schedule from equipment with auxiliary resources to cluster tools is one of the interesting topics which is required the state of the art of OR&DS techniques.
- Beside the dispatching, dynamic scheduling in semiconductor itself is a challenging topic. Scheduling system should design in a way such that consider the bottlenecks, reduce the length of production time or in another word the cycle time, maximize the throughput capacity and wafer capacity, and make a balance between the raw material inventory, wafer in process inventory, and finish product inventory.
- Run-to-run (R2R) control of semiconductor fabrication because of re-entrant flow of production process, required a flexible, accurate, stable, and fast optimization process. The main challenge is how to design the R2R control such that can deal with high-mixed dynamic scheduling plan of wafer fabrication. In addition, ITRS projected a roadmap for yield enhancement and error reduction which demanded a highly reliable control system.
- Delay for characteristics measurement from Metrology tools is unavoidable in semiconductor industry. This is a source of measurable and predictable uncertainties, however, make a challenge for process engineers to design a quality control system to deal with this source of uncertainty. Yet, there are several sources of unmeasurable uncertainties which in brief call noise. Dealing with noise is another challenge in semiconductor manufacturing environment.
- The final product in wafer fabrication is integrated circuit packaging for protecting the semiconductor device. The main challenge in this step is designing a packaging system which can protect the integrated circuit from

environmental changes like thermal effect and particle effect, or in general disturbance effect.

- Semiconductor manufacturing process is engaged with chemical processing. Most of the chemical processes are the source of uncertainties, they reduce the lifetime of fabrication equipment, are the source of particle, and change the balance in environmental factors. If the chemical process, doesn't react well for any reason, this will be affected on the quality of the wafer. One of the challenging processes which deal with chemical reaction is the etching process. The lifetime of etching tools is less than three days, and any uncertainty caused by quality reduction of etching tools affect on edge, depth, and length of the wafer, called critical dimension error.
- The automated material handling in semiconductor fabrication although brings a huge source of benefits to this industry, however, the dynamic scheduling system of wafer fabrication required a dynamic allocation system for material handling as well.

11.2.2   Review Method

The methodological review used in this study is the systematic review with the objective of history review, and status quo review [29]. In the first place, the duration of review is narrowed by the milestone of national manufacturing strategies since 2011. We abstracted how with development the national manufacturing strategies semiconductor industry is adapted to vision and evolution of the smart industry. From studies conducted after 2011, especially recent trends since 2017, most prevalent terms selected out of index terms of papers in the field of "smart semiconductor" or "semiconductor intelligence". The candidate search terms considered to be the most linked items to the scope of this paper are summarized in Table 11.1.

In this paper, the systematic review conducted based on several classification methods to categorize the review papers as follows:

- Organize the **type of research** methods by Wieringa et al. [30] (including: validation, evaluation, solution, philosophical, opinion, experience).
- classify the **areas of manufacturing** by Meziane et al. [31] (including: quality management, design, process and planning, control, environment, health and safety, maintenance and diagnosis, scheduling, and virtual manufacturing).
- categorize the **form of contribution** by keywording method [32] (including: architecture, framework, theory, methodology, model, platform, process, tool).
- classify the **type of analytic** by Delen et al. [33] (including: descriptive, predictive, and prescriptive).

11.3 OR&DS Problems in Semiconductor

As mentioned previously in Section 11.1, the main challenges and threats engaged in semiconductor manufacturing and smart industry can be answered by OR&DS perspective solutions. Following this section, we provide OR&DS role in smart semiconductor industry by answering some additional research questions in this direction.

11.3.1 By Growing the "Industry 4.0", How OR&DS Related Research Found Their Way into Semiconductor Manufacturing Intelligence?

The milestone of smart manufacturing by national perspective plans started with AMP by the US government in 2011, which indicates the timeline of our roadmap design horizon based on OR&DS. The following is the historical review of the infrastructure of smart semiconductor manufacturing aligns with the Fourth Industrial Revolution.

- **before 2011**
  Methods such as

  – data mining [34–42], artificial intelligence [43], heuristic algorithm [44–46], machine learning [47, 48], data development management [49, 50], Fuzzy logic [51], neural network [52–54], linear programming [55], statistical analysis [56, 57], optimization method [58–62], and decision analysis [63–67]

Table 11.1 Main and candidate search terms

Major terms	Minor terms
Semiconductor manufacturing	High-tech industry, integrated circuit, wafer fabrication
Smart manufacturing	Advanced manufacturing, advanced robotics, agent-based system, augmented reality, CPS, Industry 4.0, integrated manufacturing, open manufacturing, smart manufacturing, virtual factory
Data science	Artificial intelligence, big data, classification, cloud computing, clustering, data architect, data-driven technology, data management, data mining, data visualization, deep learning, IoT, machine learning, predictive modeling, statistics
Operation research	Convex optimization, decision theory, dynamic programming, forecasting, game theory, graph theory, linear programming, mathematical programming, nonlinear programming, optimization, queueing theory, soft computing

and concepts such as

– advanced manufacturing [68], intelligence manufacturing [69], Enterprise Resource Planning (ERP) [70], Overall Equipment Efficiency (OEE) [71, 72], Decision Support System (DSS) [43, 73–77], risk management [78], virtual manufacturing [79, 80], e-manufacturing [81], electronics manufacturing service [82], research and development management [83, 84], digital management [85], and "industry as a whole" [86]

have been appearing in literature to discover the challenges in semiconductor industry and moving forward to the smart manufacturing.

- **2011**
  The birth of AMP.
  Morse [87] reviewed the reputation and future of nanomanufacturing under the AMP plan.
- **2012**
  The birth of "Industry 4.0".
  The first "International Symposium on Semiconductor Manufacturing Intelligence" (ISMI) launched in Hsinchu, Taiwan [88].
- **2013**
  The first US patent [89] cited the "Industry 4.0" into semiconductor industry.
  The earliest field in order of "Industry 4.0" was in the area of soft computing for scheduling dilemma in semiconductor manufacturing [90].
- **2014**
  Digitalization of "Industry 4.0" has been discussed at AKL congress.
  "Industry 4.0" is introduced as the Fourth Industrial Revolution [91].
- **2015**
  The "Industry 4.0" points of view appeared for the first time in the theoretical and analytical researched. This trend was published in the area of the discrete event [92] and scheduling.
  SEMICON Europa 2015 hold in Germany [93] with the primary context of "Industry 4.0" of semiconductor industry, and among all the highlighted trend in semiconductor intelligence discussed in the area of:

  – "*Organization and Goals of the* "*Industry 4.0*" *Platform*"
    Five frameworks are considered to undertake the organization and structure of the "Industry 4.0": (1) reference architecture, standardization, (2) innovation and research study, (3) safety of networked systems, (4) legitimate context, and (5) labor training.
  – "*Cyber-Physical-Production-Systems at the BTU Model Factory*"
    Address the need for fast and adaptive reconfigurable approaches in production planning, logistics and "Manufacturing Execution Control" (MES) for the "Industry 4.0" platform.
  – "*The Right Security for the IoT*"
    Data security, system integrity, Intellectual Property (IP) and product and service

quality were sanctioned as the requirement for fruitful application of "Industry 4.0".

– "*Technical Visions of "Industry 4.0"*"
Explained in what ways semiconductor industry can sustain its role as the innovation driver in the area of manufacturing technologies and how it can grow from "Industry 4.0" initiative.

– "*Connecting things and services. How Industry 4.0 increases the benefit of automation at the Bosch 200mm-Waferfab*"
Showed how modularity guarantees a modest role of high-tech automation in a current environment.

– "*Interface A: Candidate for "Industry 4.0"? Adoption and Challenges in Semiconductor Industry*"
Introduced InterfaceA as an on-proprietary web technology-based interface which is equipped with data acquisition deliver a flexible interface among manufacturing tools and other IT resolutions and advances the limitation on data collection of the generic model for control of manufacturing equipment interface.

- **2016**
Following that, most industrial countries have their road map for Fourth Industrial Revolution and digitized industry, researches focused more intensely on challenges and adversities emerged with semiconductor industry and smart manufacturing. Among all, some important researches are listed as follows:

  – Dequeant et al. [94]: a comprehensive review on variability in semiconductor manufacturing to meet the "Industry 4.0" obligations.
  – Waschneck et al. [95]: a comprehensive review of job-shop scheduling. A discussion on the complexity issue with regards to the delegation of authority of decisions, tractability and adaptableness, incorporation and interacting, human aspects, and other "Industry 4.0" frustrations.
  – Moyne et al. [96]: a discussion on the requirements of data analytics, merging, quality, rates, and volumes for digitalis semiconductor industry in control process.
  – Tang et al. [97]: a discussion on the application of big data and IoT for reliability assessment in semiconductor industry.
  – Weber [98]: an introduction to the e-manufacturing on semiconductor device modeling.
  – Herding and Mönch [99]: an introduction to agent-based planning control system for semiconductor.

- **2017**
Researches have exponential growth with 100% improvement compared to 2016. Out of over 400 academic papers, the highest percentage of researches were in the field of OR (∼50%), following by DS (∼25%), roadmap and management field (∼12.5%) and image processing (∼12.5%) solutions.

- **2018**

  The Sixth ISMI was held in February in Hsinchu, Taiwan and among all the highlighted trends in semiconductor intelligence discussed in the area of:

  – *The future of smart semiconductor manufacturing*
    The remanufacturing issue will be a new topic in wafer industry; the sharing economy will enter into semiconductor industry such that customers will design the products and semiconductor manufacturers may not be known only for IC products [100].
  – *Optimization of process tool operation for future semiconductor manufacturing*
    Chamber cleaning can meet the extreme in quality control while inducing the complexity. However, optimization with reinforcement learning can reduce the complexity [101].
  – *From smart machines to smart SCs: some missing pieces*
    The term "smart" doesn't indicate of using the ICT technology to take the faster decision. Smart means: better operations management decisions (more on-time delivery, better asset utilization, less inventory, lower costs, higher quality) and better systems design decisions (faster ramp, greater flexibility, higher adaptability) [102].
  – *Manufacturing and SC optimization with "Augmented Reality" (AR) technology and "Industry 4.0" concept*
    Discussion in a thriving industry and academic collaboration for the most extensive shipbuilder in the world by integrating an optimization method and innovative IT technology, AR. They developed an advanced SC and manufacturing solution named SCM-AR based on AR and Mixed Reality solutions in collaboration with Samsung Heavy Industries Co. [103].

11.3.2 What Kind of Studies Is Being Carried out in the Field of OR&DS in Semiconductor Manufacturing?

The main objective of the above inquiry is to focus on the sort of research is being carried out in OR&DS field in terms of philosophical point of view along with practical assessments. To investigate this question, as the foremost step, Table 11.1 is
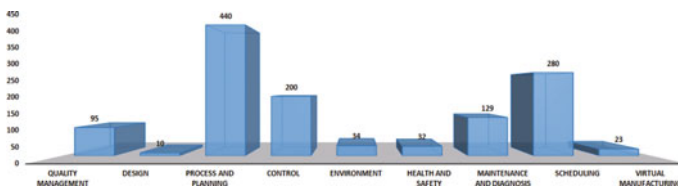

Fig. 11.1 Class allotment of **areas of manufacturing** for smart semiconductor industry

used to search all relevant articles since 2011. The articles (conference and journal papers), which were included at least one of the minor terms in semiconductor manufacturing and smart manufacturing and data science/operations research selected for the further analysis. We explored over 900 academic publications within this research. The classification result according to the definition of **areas of manufacturing** by Meziane et al. is depicted in Fig. 11.1. The result ratifies that there is an extensive gap in fitting the manufacturing design for intelligence layout. The intelligence layout design for manufacturing generally refers to system engineering design, sensor allocation problems and design the software agent solutions merge with high-tech computing technology or service-oriented computing. There is also a lack of investigation on virtual manufacturing, simulation the physical environment, e-manufacturing, and AR. In addition, trends related to the environmental issues and health and safety such as green industry and remanufacturing are demanding topics for smart manufacturing, which had less attention in semiconductor industry yet.

To determine the gap of the research for smart IC industry, we modified the classification by Meziane et al. with semiconductor manufacturing context. Some of the highlighted literature are cited as follows:

- OR

  - scheduling [104–109], production planing [110–112], job-shop scheduling [113], facility layout [114, 115], batch processing [115], bottleneck [116–118], dispatching [119–121], cycle time reduction [122–125], material handling [126], SC management [127, 128], inventory management [129, 130], demand forecasting [131, 132], capacity planing [133–138], lead time [139], supplier selection [140], purchase order [141], resource management [142, 143], pricing [144, 145], predictive maintenance [146], condition monitoring [147], operations planing and control [148–150], product quality [151], new product development [152, 153], industry development [154], user experience and interface [155–158], customer behavior[159], performance measurement [160], portfolio model [161], decision support system [162–164], large scale optimization [165], and sustainability [166].

- DS

  - Yield enhancement and prediction [167–172], WAT test [173], fault detection and classification [174–176], pattern extraction [177–179], root-cause detection [180], attribute decomposition [181], virtual metrology [182], rule-based system [183], and factor analysis [184].

Figure 11.2, illustrates the contributions of each topic in smart semiconductor industry. The scale of contribution defines such that the most relevant topic granted the smart semiconductor industry with the score of 100. Among all highlighted items, the yield enhancement and prediction, the scheduling problems, supply chain management, sustainability, and control system, are the major field of interest in articles since 2011. Due to dependency among process steps in wafer fabrication, challenges are spread along the production process such that single solution cannot clear up the
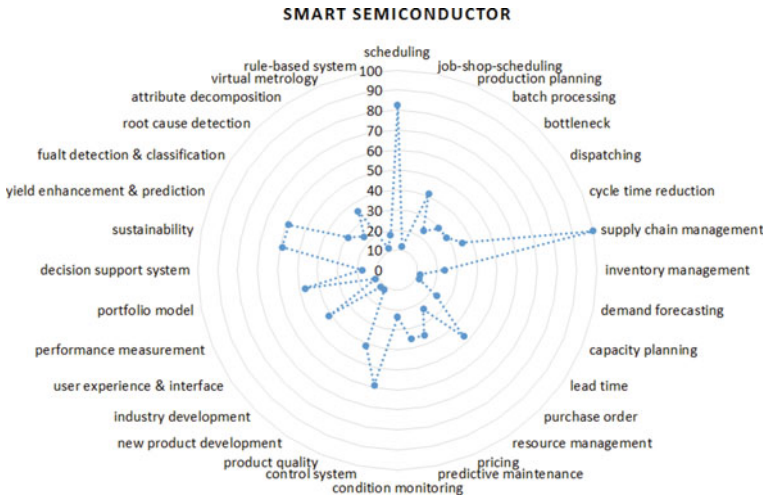
Fig. 11.2 Contribution of most frequent topics among the literature since 2011 related to smart semiconductor industry
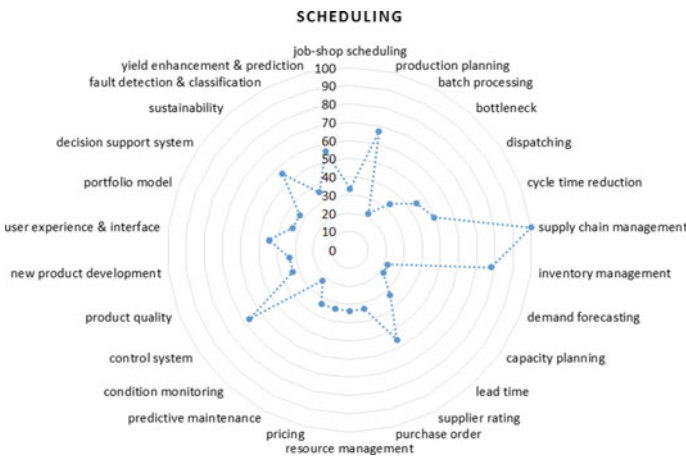

Fig. 11.3 Contribution of most frequent topics among the literature since 2011 related to scheduling in semiconductor industry

problem. Therefore, the hybrid models are a ubiquitous solution in semiconductor-related literature to deal with an epidemic dimension of problems. Figures 11.3, 11.4, 11.5, 11.6, and 11.7 demonstrate how the hybrid method is associated with each other where we only selected the most common techniques from Fig. 11.1. The results prove that the significant contribution is reminding among the most interesting topics, and there is an obligation for forming the hybrid configuration of OR&DS models for overcoming the dynamicity and measurement/unmeasurement uncertainty.
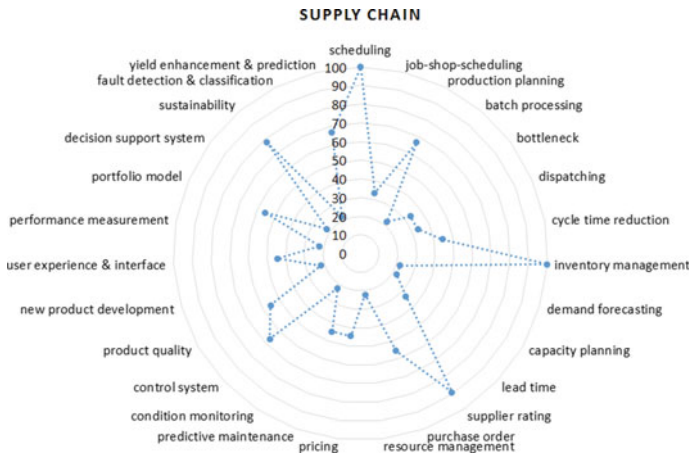
Fig. 11.4 Contribution of most frequent topics among the literature since 2011 related to supply chain in semiconductor industry
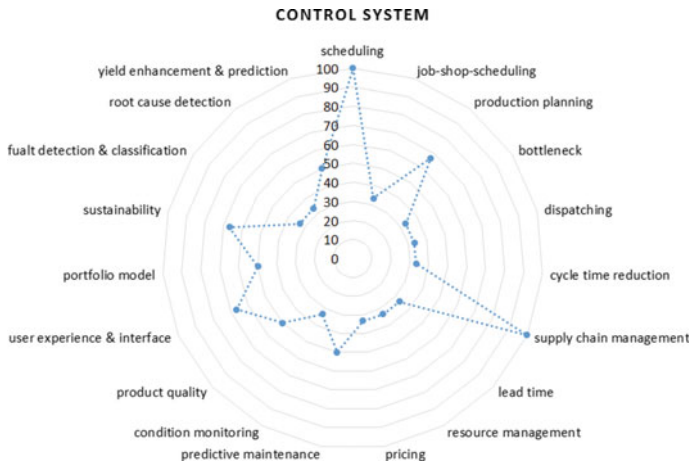

Fig. 11.5 Contribution of most frequent topics among the literature since 2011 related to control system in semiconductor industry

The classification study for the type of research method by Wieringa et al. [30] is illustrated in Fig. 11.8a, b. Figure 11.8a shows that how the type of research is branched over topics, and Fig. 11.8b shows the contribution of each type of research based on philosophical points of view. For simplicity of comparison, according to the definition of "experience" in [30], and since this type of research his seldom happen in OR&DS field, we remove the experience from the list. Concluded from Figs. 11.3, 11.4, 11.5, 11.6, 11.7, and 11.8b, the decision support system and digitization the knowledge-based system have the lowest contribution among the other research topic
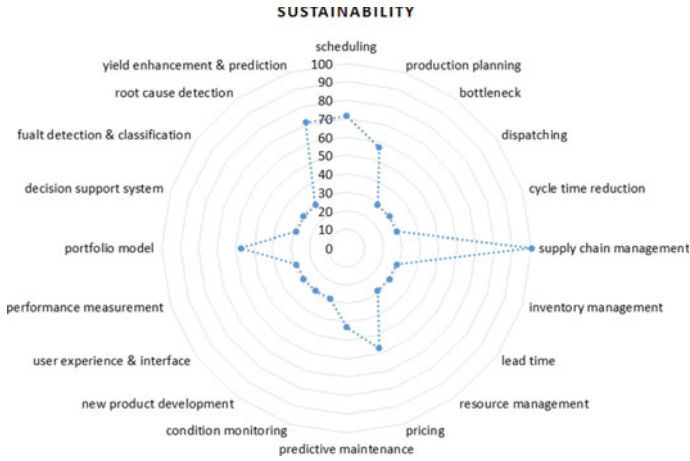
Fig. 11.6 Contribution of most frequent topics among the literature since 2011 related to sustainability in semiconductor industry
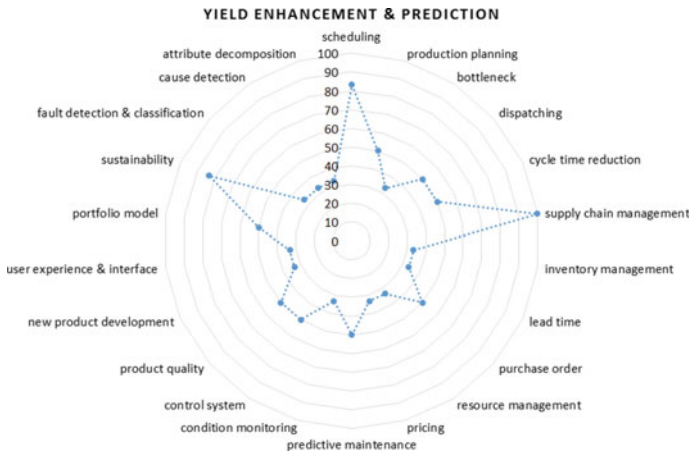

Fig. 11.7 Contribution of most frequent topics among the literature since 2011 related to yield enhancement and prediction in semiconductor industry

in current status which are required to have more inspection for advance development of smart semiconductor industry.
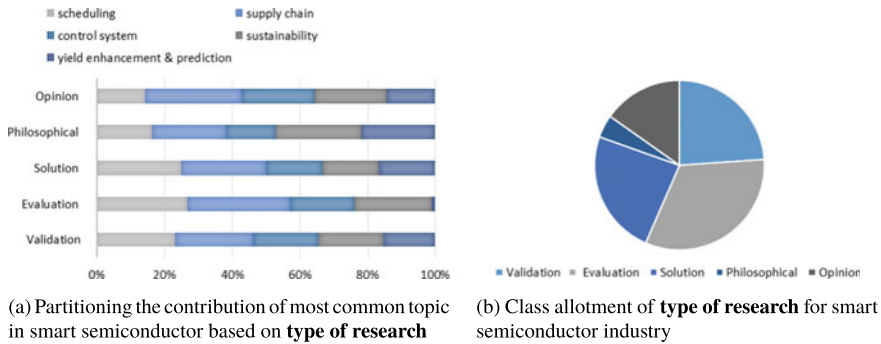
(a) Partitioning the contribution of most common topic in smart semiconductor based on **type of research**

(b) Class allotment of **type of research** for smart semiconductor industry

Fig. 11.8 Classification study

11.3.3 Which Areas of Semiconductor Manufacturing Are OR&DS Techniques Being Applied In?

The objective of the above inquiry is to highlight the types of inputs and outcomes from research struggles in the field of OR&DS. To categorize the literature according to the form of their contributions [32], we divided the attributes of contributions into two groups of variability on outcome and result (including architecture, framework, model, methodology), and variability on input information (including theory, platform, process, tool). In this category, platform indicates to the hardware or software components which enable the applications to execute while framework is the software solution for the problem. The process is the low-level processes to overcome the solution for problem and methodology is the approach to reach to that solution. The theory is the guideline or roadmap for entering to the mathematical model. Subsequently, the tool addresses to the utilities for proposing the solution, and architecture is components which interact together to achieve the solution. Figure 11.9, illustrates the 2D plot between each category. The result shows that there is a vacancy for research on integration the mathematical model with software utilities, and hardware platforms. In addition, barely the mathematical solution has been used as the roadmap for decision makers which can be investigated in the future. The theoretical approaches for developing the smart semiconductor industry plus compatible utilities with high-tech computing technology have opportunely for further study.

11.3.4 What Kind of Analytical Analysis Is Being Used in the Area of OR&DS in Semiconductor Manufacturing?

The objective of the above inquiry is to discuss the analytics of OR&DS in the study carried out to smart technologies in semiconductor industry.
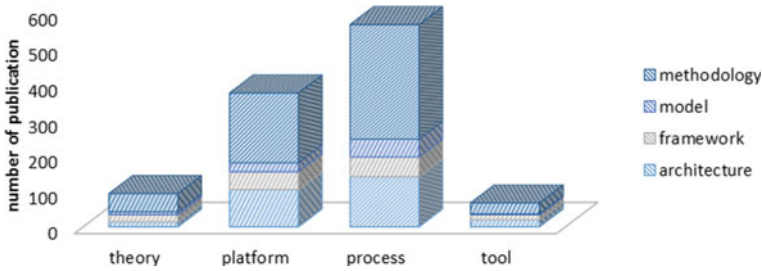
Fig. 11.9 Class allotment of **areas of manufacturing** for smart semiconductor industry

According to Delen et al. [33], the analytical analysis is classifying to descriptive, predictive, and prescriptive analysis where the descriptive analysis enables the business reporting, dashboards, data warehousing, and scorecards. Subsequently, the predictive analysis facilities data mining, forecasting, text mining and web or media mining and prescriptive analysis empower the expert systems, decision models, optimization, and simulation. Although we expect that the application of descriptive analysis and Web mining or text mining in semiconductor manufacturing is sporadic, we still considered all aspects of analytical analysis. The level of interest of each class of taxonomy presented by Fig. 11.10. Apparently, for advancement, the smartness into semiconductor industry, the descriptive analysis it will be an inevitable implement mainly for visualization the production process from the event-driven process.

To concentrate more deeply on analytical methods and their applications on the semiconductor industry, we come back to challenges discussed in Section 11.2 and review how analytic approaches applied to top challenges on control process of semiconductor products. Generally speaking, the popularity of techniques strongly depends on the popularity and severity of the challenges. The following are the details of applied methods for each challenge.

- **Photolithography process, overlay error**—challenges for compensating overlay error can be investigated through image processing [185, 186] (such as deep learning, and AI solutions [187]), optimal control algorithm design (such as linear and nonlinear programming and optimization [150, 188]), and learning-based algorithm (such as Markov decision process [189]) for enhancing the performance of robots and automated devices.
- **Scheduling and dispatching**-techniques in this field are not varied, more focuses are on optimization problems, however, the objective of optimizing models make a big emphasization on researches. The general optimization techniques appear in literature are meta-heuristic approaches or integer programming [104, 106–109, 113, 139, 190–194] in regards to the complexity and nature of problems. The minor challenges are addressed the batch data processing and dealt with this phenomena by simple techniques such as linear multivariate regression [195]. Recent trends utilized the integration of scheduling and dispatching control problems with other challenges in the production process such as advanced process control or
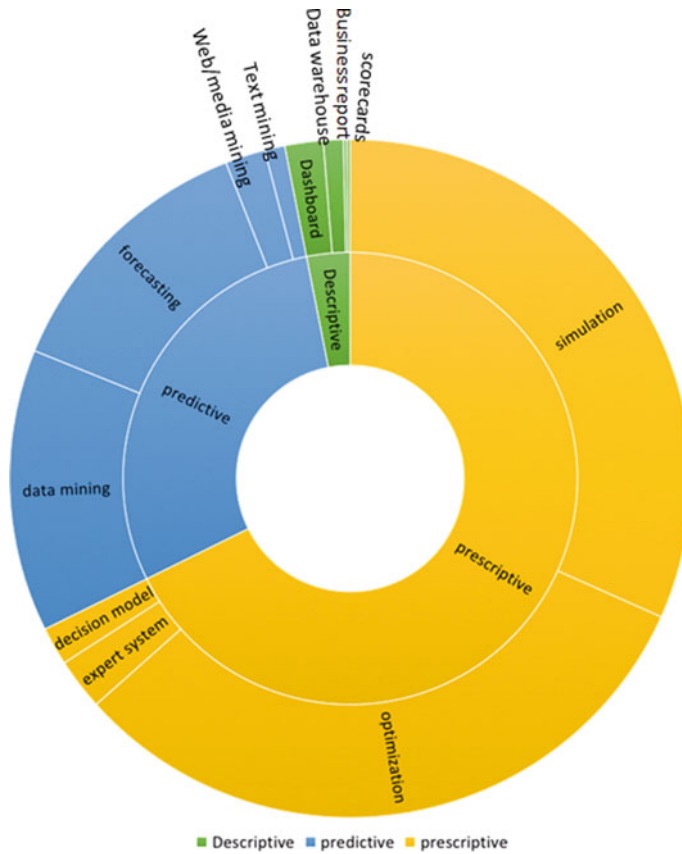
Fig. 11.10 Class allotment of **areas of manufacturing** for smart semiconductor industry

multi-agent-based system [196–198] and applied reinforcement learning for Heterogeneous sources [199–202] or AI models (e.g., neural network) [203] for the propose of clustering machine subgroups. If the challenges are related to queueing process, Markov processes also are alternative solutions in this field [204].

- **R2R control**—challenges for the R2R control in the semiconductor industry is divided into the design, optimization, application and process improvement. In the area of design, R2R controller could be designed for multi-input-multi-output (MIMO), multi-input-single-output (MISO) and single-input-single-output (SISO) systems, regards how is the dependency issue among the control variables [150, 168, 182, 205]. The most general structure of the R2R controller in the semiconductor industry is exponentially weighted moving average algorithm (EWMA) [206]. Consider the investigated problem and complexity of the control system, the EWMA is adjusted to double-EWMA (for multi-stage tasks) [207, 208] or threaded-EWMA (for the mixed process) [209]. In addition, smarter

algorithms are intended to utilize optimal control design and replace it with traditional and popular EWMA techniques, recommended solutions are kernel-based algorithm [210], proportional–integral–derivative controller (PID) [150], learning-based algorithm [211], and game theoretical approaches (e.g., mini-max optimization) [171]. Trends in this field has potential to integrate with other techniques to deal with the different source of complexity in wafer fabrication process, such as clustering or principal component-based model for identifying the similar batch and design the decentralized R2R control for batch processing systems [212, 213]. The other major sources of challenges emerged with R2R control are addressing the improvement the control process through auxiliary resources, such as stability of control system under different sources of uncertainties and reach to steady-state control design [214–216], introduce new indexes for measuring controllability and reproducibility [211], enhance the precision of control parameters by automated or self-tuning algorithm [217], and emerge the control plant with metrology equipment [210, 218–220]. In addition, the R2R controller can assist with other phenomena rather than a control process, including, fault detection and root-cause identification and classification [221], change point detection, and yield enhancement [171]. Therefore, consider the application, the analytical approaches are specified (for details about model selection refers to other challenges). Other notable questions are how to deal with small size of data, how to derive error smoothly, how to design the R2R control system for dynamic system, and how to consider within process and between process R2R control models.

- **Disturbances and delay**—the main purpose of researches in this field is an investigation on the stability of other solution on the presence of any source of uncertainties. To do that, the first step is to simulate the unmeasurable uncertainties, the highlighted techniques are generally based on virtual metrology tools and can be classified on rule-based models such as fuzzy systems [222], or stochastic processes (e.g., Gaussian and non-Gaussian processes) [223]. In addition, plans for efficient and proper sampling can enhance the quality of data and reduce the noise [224]. Trends for root-cause detection and classification can help to find the source of uncertainties, the most approachable methods in this field are data mining approaches (e.g., principal component analysis, neural network or in general clustering or classification techniques) [225, 226]. In general words, challenges are tightened up with disturbance rejection models basically are related in control design and algorithm [227, 228].
- **Packaging**—since the quality of packaging strongly depends on thermal effects, major researches are addressed this challenges through the reliability and survival analysis such as degradation models or accelerated test [229]. In addition, before testing the reliability of packaging the thermal effects are predictable by thermal models based on Fourier series or Kalman filtering models [230]. To investigate the quality of packaging, few studies indicated this phenomenon by image processing [231, 232]. Furthermore, the packaging is almost the final production process in wafer fabrication, therefore has a strong correlation with yield testing result. Therefore, for the yield management purpose, one solution is to conduct the root-cause detection and classification based on the result from the packag-

ing process. Applicable methods in this step are clustering techniques with data mining approaches [40, 233–235].

- **Critical dimension**—although most approaches in this field are related to chemical and nanoengineering area, yet from the OR&DS perspective, the challenges for critical dimension enhancement are investigated by advanced process control and statistical quality control approaches [236, 237]. However, the recent advanced trends in this field are included the hybrid algorithm which could increase the performance of controller by adding optimization into process control [238, 239]. The more intelligent hybrid techniques are combined with the sequential learning or kernel-based learning such as support vector regression or other machine learning methods [167, 240, 241]. Some researches intended to produce virtual data by virtual metrology tools [242] to be able to have enough data as the basis requirements for applying traditional statistical inferences combined with learning techniques such as LASO and ridge regression [243, 244]. Other trends are investigated on root-cause identification for identifying the source of uncertainties and environmental protection through data mining approaches [171].
- **Scheduling for automated material handling**—scheduling challenges can be cover by two approaches, first, design an intelligent scheduling system basically through queueing theory and stochastic process [245], design the distributed network system by mathematical modeling languages such as Petri net [246–248], design the facility allocation by simulation optimization or design of experiment techniques such as Taguchi method [249, 250]. Second, find the optimal performance of dynamic multi-objective scheduling design through approaches such as heuristic optimization [143], sequencing optimization [251], and combinatorial optimization (e.g., Hungarian algorithm) [252]. The performance of the scheduling system could be measured in the field of quality control.

11.4 Management Suggestion

In semiconductor, managers need to overcome different challenges which are being mentioned in the above sections. Despite those challenges, in the following, we give certain future circumstances to "Industry 4.0" standpoints.

Digitalize knowledge-based decision support system

- Incorporating the behavior of human decision makers with proposed solutions.
- Automating decisions made by humans.
- Highlighting the interface of information systems with humans

Incorporate the dynamicity into the solutions

- Developing stochastic and dynamic versions of solutions and deterministic models.
- Anticipating the stochasticity in the models based on dynamic programming, robust optimization, and stochastic programming.

Design software-based solution with user-friendly interface

- Considering the role of high-tech computing techniques including cloud computing techniques in decision-making and parallel computing on Graphics Processing Units (GPU).
- Knowing the restrictions of current packaged software for semiconductor management, process, and production.
- Proposing alternative software solutions including service-oriented computing and software agents for semiconductor planning and scheduling applications.

Forming the hybrid configuration of OR&DS models

- Facilitating planning problems and decision-making-based OR perspective by data mining techniques.
- Implementing "Manufacturing Execution System" (MES), "Enterprise Resource Planning" (ERP), and "Advanced Planning and Scheduling" (APS) for developing the integrated production planning and scheduling solutions.
- Decreasing the measurement uncertainty by merging the hybrid metrology with state-of-the- art statistical analyses [253].

Simulation and data-driven solutions

- Simulating the physical environment in order to comprehend the connections amid the real setting circumstance and planning to find solution approaches in the risk-free environment before applying them.
- Visualizing production planning processes by the use of the event-driven process.
- Modeling and analyzing semiconductor challenges by utilization of various simulation paradigms (i.e., agent-based systems, hybrid models, reduced simulation models, systems dynamics).
- Supporting the different aspect of decision-making in semiconductor by embedding the actual simulation methods in existing and forthcoming information systems.

Process integration

- Integrating decisions made by the different elements in the system to avoid the ad hoc situation.
- Integrating the high-tech computing procedures to derive the computationally tractable models, and to discourse the diverse uncertainties come across in the industry [254].
- Incorporating sustainability aspects into proposed solutions and deterministic models.
- Integrating the product lifetime into account for demand planning [255].

11.5 Conclusion and Future Research Direction

As a conclusion and future research direction, we attempted to have a broader vision on the requirements for industrial development and intelligence manufacturing of semiconductor products. These requirements are barely indicated in literature with

analytic context and are known as the new obligations for the next step toward smart manufacturing. In the following, we discuss some of the highlighted topics in this chain.

11.5.1 Supply Chain Management

Supply chain (SC) is growing exponentially and contributing substantially to the global economy. This growth is accompanied by continuous technology migration and minimizing cost for different applications in green energy, communication, computers, automotive, medical, and electronics industries [110]. There are some survey papers on SC in literature with the scope of needs, practices and integration issues in [256]; (1) Research agenda framework for supply network integration (questionnaire-based) in [257]; (2) Decision paradigms for SC management (questionnaire-based) in [258]; (3) Successes and opportunities in modeling and integrating planning, scheduling, equipment configuration and fab capability assessment in [259, 260]; (4) E-markets and SC collaboration in [261], and (5) Strategic SC network design and SC simulation models in [262–264].

According to [265] and [262], one future direction of semiconductor industry would be global SC simulation models based on a marketing operations perspective, which lead another research direction in the area of operations management such as production planning and demand fulfillment, inventory control, capacity and demand planning, and marketing and sales models. Moreover, positioning the "Order Penetration Points" (OPPs) in global semiconductor SC networks is another strategic competitive decision, especially for novel product architectures with new options which can be modeled with game theory (see [265, 266]).

11.5.2 Sustainability and Remanufacturing

Materials, products, and processes are becoming smarter, sustainable, energy aware, and innovation driven. Sustainability includes (1) Lower use of energy and materials, (2) Greater environmental friendliness [267], and (3) Circular economy and remanufacturing [18]. Nowadays, semiconductor industry has significantly and exponentially increased the rate of e-waste in daily life [268, 269]. There is a challenge for inventing efficient and pollution-free high-tech recycling technologies for e-waste which help to enhance the comprehensive utilization of resources, and consequently, it will develop the cyclic economy. There is a critical future research direction on new recycling electrostatic separation, which is simple and optimize energy consumption without any wastewater discharge to recover the mixtures containing conductors (copper), semiconductors (extrinsic silicon), and nonconductors (woven glass reinforced resin) in semiconductor [270].

11.5.3 Green Smart Semiconductor Manufacturing

Another future research stream would be data-driven decision-making and optimization applications in integrated Smart and Green Manufacturing. Some application challenges in this area would be: (1) Business Model Challenge: manufacturers face threats from digital disruptors that are often quicker to adapt traditional products and exploit new opportunities through the latest technology. (2) Data and Security Challenge: Smart manufacturing is heavily reliant on technology and data which brings with it the challenges of protecting that data and ensuring it is secure. Smart manufacturing systems and the generated data from that might also be targets for cyber attacks. (3) Operations Challenges: Manufacturers need to be agile and respond more quickly to update their technology. Connecting different systems to get an end-to-end picture of the manufacturing process, supply chain, and product usage are a further challenge [271].

Eventually, the fast-growing semiconductor manufacturing requires a Knowledge Management Systems (KMS) in order to support management DSS. This KMS will identify and analyze research trend gaps and organize a future research agenda for new product development [272].

References

1. Advanced manufacturing partnership. https://www.manufacturing.gov
2. Kagermann, H., Lukas, W.-D., Wahlster, W.: VDI nachrichten. **13**, 11 (2011)
3. la nouvelle france industrielle. https://www.economie.gouv.fr/entreprises/nouvelle-france-industrielle
4. Future of manufacturing. https://www.gov.uk/government/collections/future-of-manufacturing
5. Ersson, C., Sagström, E.: Made in Sweden 2030 (2013)
6. Factories of the future. http://ec.europa.eu/research/industrial_technologies/factories-of-the-future_en.html
7. Kang, H.S., Lee, J.Y., Choi, S., Kim, H., Park, J.H., Son, J.Y., Kim, B.H., Do Noh, S.: Int. J. Precis. Eng. Manuf. Green Technol. **3**(1), 111–128 (2016)
8. Industria conectada 4.0. http://www.industriaconectada40.gob.es/
9. Smart Industry Netherlands. https://www.smartindustry.nl/english/
10. Industry 4.1j. http://www8.cao.go.jp/cstp/english/basic/
11. Made in China. https://www.csis.org/analysis/made-china-2025
12. Fabbrica Intelligente of Italy. http://www.fabbricaintelligente.it/en/
13. Innovation and enterprise Singapore. https://www.nrf.gov.sg/rie2020
14. Baheti, R., Gill, H.: Impact Control Technol. **12**, 161–166 (2011)
15. Gershenfeld, N., Krikorian, R., Cohen, D.: Sci. Am. **291**(4), 76–81 (2004)
16. Modha, D.S., Ananthanarayanan, R., Esser, S.K., Ndirango, A., Sherbondy, A.J., Singh, R.: Commun. ACM **54**(8), 62–71 (2011)
17. Hayes, B.: Commun. ACM **51**(7), 9–11 (2008)
18. Kusiak, A., et al.: Nature **544**(7648), 23–25 (2017)
19. Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., Hoffmann, M.: Bus. Inf. Syst. Eng. **6**(4), 239–242 (2014)

20. Fathi, M., Khakifirooz, M., Pardalos, P.M.: Optimization in Large Scale Problems: Industry 4.0 and Society 5.0 Applications, vol. 152. Springer Optimization and Its Applications (2019)
21. Schaller, R.R.: IEEE Spectr. **34**(6), 52–59 (1997)
22. Arden, W., Brillouët, M., Cogez, P., Graef, M., Huizing, B., Mahnkopf, R.: Version **2**, 14 (2010)
23. International technology roadmap for semiconductors. http://www.itrs2.net
24. International technology roadmap for semiconductor 2.0 edition 2015: more Moore. https://www.semiconductors.org/clientuploads/research_technology/itrs/2015/5_2015%20itrs%202.0_more%20moore.pdf
25. International technology roadmap for semiconductors 2.0 edition 2015: factory integration. https://www.semiconductors.org/clientuploads/research_technology/itrs/2015/7_2015%20itrs%202.0%20factory%20integration.pdf
26. Khakifirooz, M., Fathi, M., Wu, K.: Development of smart semiconductor manufacturing: operations research and data science perspectives. In: IEEE Access, vol. 7, pp. 108419–108430 (2019). https://doi.org/10.1109/ACCESS.2019.2933167
27. Mönch, L., Fowler, J.W., Mason, S.J.: Production Planning and Control for Semiconductor Wafer Fabrication Facilities: Modeling, Analysis, and Systems, vol. 52. Springer Science & Business Media (2012)
28. Hall, B.H., Ziedonis, R.H.: Rand J. Econ. 101–128 (2001)
29. Noguchi, J.: The Science Review Article: An Opportune Genre in the Construction of Science, vol. 17. Peter Lang (2006)
30. Wieringa, R., Maiden, N., Mead, N., Rolland, C.: Requir. Eng. **11**(1), 102–107 (2006)
31. Meziane, F., Vadera, S., Kobbacy, K., Proudlove, N.: Integr. Manuf. Syst. **11**(4), 218–238 (2000)
32. Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: In: EASE, vol. 8, pp. 68–77 (2008)
33. Delen, D., Demirkan, H.: Data, Information and Analytics as Services. Elsevier (2013)
34. Chien, C.-F., Wang, W.-C., Cheng, J.-C.: Expert Syst. Appl. **33**(1), 192–198 (2007)
35. Hsu, S.-C., Chien, C.-F.: Int. J. Prod. Econ. **107**(1), 88–103 (2007)
36. Chien, C.-F., Chen, L.-F.: Expert Syst. Appl. **34**(1), 280–290 (2008)
37. Chien, C.-F., Hsiao, C.-W., Meng, C., Hong, K.-T., Wang, S.-T.: In: IEEE International Symposium on Semiconductor Manufacturing, 2005, ISSM 2005, pp. 327–330. IEEE (2005)
38. Chien, C.-F., Lin, T.-H., Peng, C.-Y., Hsu, S.-C.: J. Chin. Inst. Ind. Eng. **18**(4), 37–48 (2001)
39. Chien, C.-F., Hsiao, A., Wang, I.: J. Chin. Inst. Ind. Eng. **21**(4), 313–327 (2004)
40. Chien, C.-F., Li, H.-C., Jeang, A.: Intell. Syst. Account. Financ. Manag. **14**(1–2), 43–57 (2006)
41. Hsu, C.-Y., Chien, C.-F., Lin, K.-Y., Chien, C.-Y.: J. Chin. Inst. Ind. Eng. **27**(2), 140–156 (2010)
42. Chien, C.-F., Lin, T.-H., Liu, Q.-W., Peng, C.-Y., Hsu, S.-C., Huang, C.-C.: J. Chin. Inst. Ind. Eng. **19**(2), 23–38 (2002)
43. Wang, K.-J., Chien, C.-F.: Robot. Comput. Integr. Manuf. **19**(1–2), 65–77 (2003)
44. Chien, C.-F., Chen, C.-H.: OR Spectr. **29**(3), 391–419 (2007)
45. Chien, C.-F., Hsu, S.-C., Peng, S., Wu, C.-H.: In: Semiconductor Manufacturing Technology Workshop, 2000, pp. 217–229. IEEE (2000)
46. Chen-Fu, C., Chien-Hung, C.: OR Spectr. **29**(3), 391 (2007)
47. Chien, C.-F., Chen, L.-F.: IEEE Trans. Semicond. Manuf. **20**(4), 528–541 (2007)
48. Lin, K.-S., Chien, C.-F.: Expert Syst. Appl. **36**(2), 3327–3335 (2009)
49. Leachman, R.C., Ding, S., Chien, C.-F.: IEEE Trans. Autom. Sci. Eng. **4**(4), 501–512 (2007)
50. Lin, W.-L., Chien, C.-F., Chen, W.-C., Wu, W.-C., Wang, H.-Y., Kuo, R.-T., Chou, M.-H.: In: 2008 International Symposium on Semiconductor Manufacturing (ISSM), pp. 125–128. IEEE (2008)
51. Wang, K.-J., Lin, Y.-S., Chien, C.-F., Chen, J.: Robot. Comput. Integr. Manuf. **25**(1), 32–41 (2009)
52. Mashiko, K.: Semiconductor neural network including photosensitive coupling elements, US Patent 4,988,891 (1991)
53. Chen, F.-L., Liu, S.-F.: IEEE Trans. Semicond. Manuf. **13**(3), 366–373 (2000)

54. Su, C.-T., Yang, T., Ke, C.-M.: IEEE Trans. Semicond. Manuf. **15**(2), 260–266 (2002)
55. Wu, J.-Z., Chien, C.-F.: OR Spectr. **30**(3), 401–430 (2008)
56. Peng, C.-Y., Chien, C.-F.: Int. J. Serv. Technol. Manag. **4**(4–6), 365–383 (2003)
57. Chen, P.-N., Chien, C.-F., Wang, S.-J., Chen, C.-C., Luo, H.-J.: In: Semiconductor Manufacturing Technology Workshop Proceedings, 2004, pp. 174–177. IEEE (2004)
58. Grundel, D., Murphey, R., Pardalos, P., Prokopyev, O.: Cooperative Systems: Control and Optimization, vol. 588. Springer Science & Business Media (2007)
59. Chien, C.-F., Hsu, S.-C., Deng, J.-F.: IEEE Trans. Semicond. Manuf. **14**(2), 157–162 (2001)
60. Zheng, J.-N., Chang, K.-H., Chien, C.-F.: In: 40th International Conference on Computers and Industrial Engineering (CIE), 2010, pp. 1–6. IEEE (2010)
61. Chien, C.-F., Chen, Y.-J., Peng, J.-T.: In: Winter Simulation Conference, 2008, WSC 2008, pp. 2313–2322. IEEE (2008)
62. Wu, J.-Z., Chien, C.-F., Huang, Y.-S., Huang, H.-Y.: In: 40th International Conference on Computers and Industrial Engineering (CIE), 2010, pp. 1–6. IEEE (2010)
63. Chien, C.-F., Chang, K.-H., Chen, C.-P.: In: Metrology, Inspection, and Process Control for Microlithography XV, vol. 4344, pp. 245–257. International Society for Optics and Photonics (2001)
64. Chien, C.-F., Liu, C.-C., Hsu, C.-Y., Chou, H.-S., Lin, C.-W.: In: International Symposium on Semiconductor Manufacturing, 2007, ISSM 2007, pp. 1–4. IEEE (2007)
65. Chien, C.-F., Deng, J.-F.: Int. Trans. Oper. Res. **8**(5), 535–545 (2001)
66. Wang, H.-J., Chien, C.-F., Kuo, C.-J.: Analyzing alternative strategies of semiconductor final testing. In: Multi-objective Programming and Goal Programming. Springer (2003)
67. Chien, C.-F., Chang, K.-H., Chen, C.-P.: J. Chin. Inst. Ind. Eng. **18**(3), 95–103 (2001)
68. Nijs, J.F., Szlufcik, J., Poortmans, J., Sivoththaman, S., Mertens, R.P.: IEEE Trans. Electron Devices **46**(10), 1948–1969 (1999)
69. Chien, C.-F., Chen, Y.-J., Peng, J.-T.: Int. J. Prod. Econ. **128**(2), 496–509 (2010)
70. Wei, C.-C., Chien, C.-F., Wang, M.-J.J.: Int. J. Prod. Econ. **96**(1), 47–62 (2005)
71. Chien, C.-F., Chen, H.-K., Wu, J.-Z., Hu, C.-H.: Int. J. Prod. Res. **45**(3), 509–524 (2007)
72. Chien, C.-F., Hsu, C.-Y., Chou, H.-S., Lin, C.-W.: In: IEEE International Symposium on Semiconductor Manufacturing, 2006, ISSM 2006, pp. 317–320. IEEE (2006)
73. Chien, C.-F., Wu, J.-Z.: IEEE Trans. Semicond. Manuf. **16**(4), 704–711 (2003)
74. Chien, C.-F., Chen, W.-C., Hsu, S.-C.: Int. J. Prod. Res. **48**(23), 6959–6976 (2010)
75. Chien, C.-F., Hu, C.-H., Lin, C.-Y.: Int. J. Manuf. Technol. Manag. **14**(1–2), 130–144 (2008)
76. Chien, C.-F., Wu, J.-Z., Weng, Y.-D.: Flex. Serv. Manuf. J. **22**(1–2), 109–139 (2010)
77. Wu, H.-H., Chien, C.-F., Huang, Y.-L., Huang, D.-S., Tsai, S.-D., Huang, M.-P.: In: Proceedings of 6th Electronics Packaging Technology Conference, 2004. EPTC 2004, pp. 81–86. IEEE (2004)
78. Chen, C.-C., Chiang, Y.-S., Chien, C.-F.: In: International Symposium on Semiconductor Manufacturing, 2007, ISSM 2007, pp. 1–3. IEEE (2007)
79. Chien, C.-F., Chen, W.-C., Hsu, S.-C.: In: Winter Simulation Conference, 2008, WSC 2008, pp. 2201–2208. IEEE (2008)
80. Chien, C.-F., Wu, J.-Z.: In: IEEE International Conference on Automation Science and Engineering, 2007, CASE 2007, pp. 265–269. IEEE (2007)
81. Chang, P.-L., Chien, C.-F.: Int. J. Serv. Technol. Manag. **4**(4–6), 323–330 (2003)
82. Chien, C.-F., Shi, Y.: Int. J. Bus. **9**(4), 327 (2004)
83. Chien, C.-F., Hu, C.-H.: In: IEEE International Symposium on Semiconductor Manufacturing, 2006, ISSM 2006, pp. 265–268. IEEE (2006)
84. Chien, C.-F., Wang, J., Chang, T.-C., Wu, W.-C.: In: International Symposium on Semiconductor Manufacturing, 2007, ISSM 2007, pp. 1–4. IEEE (2007)
85. Chien, C.-F., Hsu, S.-C., Hsu, C.-Y.: Recent Adv. Data Min. Enterp. Data: Algorithms Appl. **6**, 367 (2008)
86. Chien, C.-F.: In: 2010 40th International Conference on Computers and Industrial Engineering (CIE), pp. 1–2. IEEE (2010)
87. Morse, J.: NNN Newslett. **4**(11) (2011)

88. In: International Symposium on Semiconductor Manufacturing Intelligence. https://ismi2018.decisions.org.tw

89. Assaf, S., Babu, V., Flores, R.D., Hickey, B., Kuttannair, K., Park, S.J., Rhee, A., Wang, C.C.: Method and system for managing process jobs in a semiconductor fabrication facility, US Patent 8,527,080 (2013)

90. Chen, T., Wang, Y.-C.: Int. J. Manuf. Res. **8**(2), 150–170 (2013)

91. Bloem, J., Van Doorn, M., Duivestein, S., Excoffier, D., Maas, R., Van Ommeren, E.: Things to Tighten the Link Between IT and OT (2014)

92. Fowler, J.W., Mönch, L., Ponsignon, T.: Int. J. Ind. Eng. **22**(5) (2015)

93. Semicon Europa (2015). http://www.semiconeuropa.org/bestof2015

94. Dequeant, K., Vialletelle, P., Lemaire, P., Espinouse, M.-L.: In: Proceedings of the 2016 Winter Simulation Conference, pp. 2598–2609. IEEE Press (2016)

95. Waschneck, B., Altenmüller, T., Bauernhansl, T., Kyek, A.: In: SAMI iKNOW (2016)

96. Moyne, J., Samantaray, J., Armacost, M.: IEEE Trans. Semicond. Manuf. **29**(4), 283–291 (2016)

97. Tang, T.J., Chung, A., Zhao, A., Kang, R., Zhang, M., Chien, K., Yang, J., Zhang, J.: In: 2016 2nd International Conference on Cloud Computing and Internet of Things (CCIOT), pp. 111–114. IEEE (2016)

98. Weber, A.: In: e-Manufacturing and Design Collaboration Symposium (eMDC), 2016, pp. 1–3. IEEE (2016)

99. Herding, R., Mönch, L.: In: OTM Confederated International Conferences on the Move to Meaningful Internet Systems, pp. 65–75. Springer (2016)

100. Intelligent Systems Laboratory, The University of Iowa. https://research.engineering.uiowa.edu/kusiak/

101. System Integration and Modeling Lab., Korea Advanced Institute of Science and Technology. http://simlab.kaist.ac.kr/

102. The William M. Keck Virtual Factory Lab—Georgia Tech. https://factory.isye.gatech.edu

103. System Design & Management Lab-Korea Advanced Institute of Science and Technology. http://sdm.kaist.ac.kr/wordpress/korean/

104. Chamnanlor, C., Sethanan, K., Chien, C.-F., Gen, M.: Ind. Eng. Manag. Syst. **12**(4), 306–316 (2013)

105. Zheng, J.-N., Chien, C.-F.: In: 2013 IEEE International Conference on Automation Science and Engineering (CASE), pp. 1034–1039. IEEE (2013)

106. Hao, X.-C., Wu, J.-Z., Chien, C.-F., Gen, M.: J. Intell. Manuf. **25**(5), 867–879 (2014)

107. Chou, C.-W., Chien, C.-F., Gen, M.: IEEE Trans. Autom. Sci. Eng. **11**(3), 692–705 (2014)

108. Chamnanlor, C., Sethanan, K., Chien, C.-F., Gen, M.: Int. J. Prod. Res. **52**(9), 2612–2629 (2014)

109. Wang, H.-K., Chien, C.-F., Gen, M.: In: 2014 IEEE International Conference on Automation Science and Engineering (CASE), pp. 101–106. IEEE (2014)

110. Chien, C.-F., Dauzère-Pérès, S., Ehm, H., Fowler, J.W., Jiang, Z., Krishnaswamy, S., Lee, T.-E., Moench, L., Uzsoy, R.: Eur. J. Ind. Eng. 4 **5(3)**, 254–271 (2011)

111. Chien, C.-C., Hsu, C.-Y., Chiou, N., Chien, C.-F., Hsin, W.-M., Lee, C.-Y., Chien, J., Wu, A.: In: 2011 International Symposium on Semiconductor Manufacturing (ISSM) and e-Manufacturing and Design Collaboration Symposium (eMDC), pp. 1–11. IEEE (2011)

112. Jamrus, T., Chien, C.-F., Gen, M., Sethanan, K.: Fuzzy Optim. Decis. Mak. **14**(3), 265–287 (2015)

113. Chamnanlor, C., Sethanan, K., Gen, M., Chien, C.-F.: J. Intell. Manuf. **28**(8), 1915–1931 (2017)

114. Pan, X., Li, L., Chen, Z., Jia, P.: In: The 26th Chinese Control and Decision Conference (2014 CCDC), pp. 4019–4024. IEEE (2014)

115. Chien, C.-F., Hsu, C.-Y., Chen, Y.-J.: Method of dispatching semiconductor batch production, US Patent 9,513,626 (2016)

116. Bang, J.-Y., Kim, Y.-D.: Comput. Oper. Res. **38**(3), 666–673 (2011)

117. Yang, K., Chung, Y., Kim, D., Park, S.C.: Korean J. Comput. Des. Eng. **19**(3), 214–223 (2014)

118. Deng, J., Cao, Z., Liu, M.: In: 2014 IEEE 11th International Conference on Networking, Sensing and Control (ICNSC), pp. 58–63. IEEE (2014)
119. Hildebrandt, T., Goswami, D., Freitag, M.: In: Proceedings of the 2014 Winter Simulation Conference, pp. 2580–2590. IEEE Press (2014)
120. Li, L., Sun, Z., Zhou, M., Qiao, F.: IEEE Trans. Autom. Sci. Eng. **10**(2), 354–364 (2013)
121. Wang, C.-N., Chen, L.-C.: J. Intell. Manuf. **23**(5), 2047–2056 (2012)
122. Kuo, C.-J., Chien, C.-F., Chen, J.-D.: IEEE Trans. Autom. Sci. Eng. **8**(1), 103–111 (2011)
123. Chien, C.-F., Hsu, C.-Y., Hsiao, C.-W.: J. Intell. Manuf. **23**(6), 2281–2294 (2012)
124. Hsieh, L.Y., Chang, K.-H., Chien, C.-F.: Int. J. Prod. Res. **52**(10), 3097–3109 (2014)
125. Xie, Y., Chien, C.-F., Tang, R.-Z.: Comput. Ind. Eng. **99**, 401–414 (2016)
126. Chien, C.-F., Hu, C.-H., Hu, Y.-F.: IEEE Trans. Semicond. Manuf. **29**(3), 239–247 (2016)
127. Liu, X., Pei, J., Liu, L., Cheng, H., Zhou, M., Pardalos, P.M.: Dynamic coordinated supply chain scheduling in an IoT environment. In: Optimization and Management in Manufacturing Engineering, pp. 63–90. Springer (2017)
128. Chien, C.-F., Ehm, H., Fowler, J., Mönch, L.: In: Dagstuhl reports, vol. 6-2. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2016)
129. Wu, J.-Z., Yu, H.-C., Chien, C.-F.: In: Proceedings of the 2014 Winter Simulation Conference, pp. 2591–2599. IEEE Press (2014)
130. Wu, J.-Z., Chien, C.-F., Tsou, Y.-C.: In: 2014 IEEE International Conference on Automation Science and Engineering (CASE), pp. 613–618. IEEE (2014)
131. Hsu, C.-Y., Chien, C.-F.: In: Proceedings of 12th Asia Pacific Industrial Engineering & Management Systems Conference (APIEMS 2011) (2011)
132. Chien, C.-F., Wu, J.-Z., Wu, C.-C.: Flex. Serv. Manuf. J. **25**(3), 286–309 (2013)
133. Chen, W.-C., Chien, C.-F.: Int. J. Prod. Res. **49**(12), 3635–3652 (2011)
134. Chien, C.-F., Zheng, J.-N.: J. Intell. Manuf. **23**(6), 2151–2159 (2012)
135. Chien, C.-F., Wu, C.-H., Chiang, Y.-S.: Int. J. Prod. Econ. **135**(2), 860–869 (2012)
136. Chien, C.-F., Kuo, R.-T.: Flex. Serv. Manuf. J. **25**(3), 310–342 (2013)
137. Lee, C.-Y., Chen, C.-H., Chien, C.-F.: Int. J. Prod. Res. **52**(6), 1868–1885 (2014)
138. Chien, C.-F., Dou, R., Fu, W.: Appl. Soft Comput. (2017)
139. Wang, H.-K., Chien, C.-F., Gen, M.: IEEE Trans. Semicond. Manuf. **28**(3), 353–366 (2015)
140. Hsu, C.-Y., Lin, S.-C., Chien, C.-F.: J. Ind. Prod. Eng. **32**(3), 149–161 (2015)
141. Zulkifli, N., Sivalingam, S., et al.: In: 2015 International Symposium on Technology Management and Emerging Technologies (ISTMET), pp. 325–329. IEEE (2015)
142. Chien, C.-F., Chu, P.-C., Zhao, L.: Int. J. Ind. Eng. **22** (5) (2015)
143. Chien, C.-F., Chou, C.-W., Yu, H.-C.: IEEE Trans. Autom. Sci. Eng. **13**(4), 1567–1580 (2016)
144. Zhou, Z., Liu, X., Pei, J., Pardalos, P.M., Cheng, H.: J. Ind. Manag. Optim. **13**(5), 1–16 (2017)
145. Chen-fu, C., Ren-tsun, K.: Ind. Eng. J. **2**, 000 (2013)
146. Moyne, J., Iskandar, J.: Processes **5**(3), 39 (2017)
147. Yu, H.-C., Lin, K.-Y., Chien, C.-F.: J. Intell. Manuf. **25**(5), 933–943 (2014)
148. Pardalos, P.M.: Optimization and Management in Manufacturing Engineering, vol. 126. Springer (2017)
149. Chien, C.-F., Gen, M., Shi, Y., Hsu, C.-Y.: J. Intell. Manuf. **25**(5), 845–847 (2014)
150. Chien, C.-F., Chen, Y.-J., Hsu, C.-Y., Wang, H.-K.: IEEE Trans. Autom. Sci. Eng. **11**(2), 473–484 (2014)
151. Chien, C.-F., Hsu, C.-Y., Chang, K.-H.: Comput. Ind. Eng. **65**(1), 117–127 (2013)
152. Chien, C.-F., Lin, K.-Y., Yu, A.P.-I.: Comput. Ind. Eng. **73**, 75–84 (2014)
153. Kerh, R., Chien, C.-F., Lin, K.-Y.: Int. J. Mech. Aerosp. Ind. Mechatron. Eng. **8**(4), 661–666 (2014)
154. Chen, C.-P., Chien, C.-F., Lai, C.-T.: Innovation **15**(4), 416–436 (2013)
155. Chien, C.-F., Zheng, J.-N., Lin, Y.-J.: J. Intell. Manuf. **25**(5), 899–911 (2014)
156. Chien, C.-F., Kerh, R., Lin, K.-Y., Yu, A.P.-I.: Comput. Ind. Eng. **99**, 162–173 (2016)
157. Lin, K.-Y., Chien, C.-F., Kerh, R.: Comput. Ind. Eng. **99**, 487–502 (2016)
158. Lin, K.-Y., Yu, A.P.-I., Chu, P.-C., Chien, C.-F.: J. Ind. Prod. Eng. **34**(7), 504–519 (2017)

159. Chien, C.-F., Hsu, C.-Y., Lin, S.-C.: Manufacturing intelligence to forecast the customer order behavior for vendor managed inventory. In: Intelligent Decision Technologies, pp. 51–60. Springer (2012)
160. Chen, W.-C., Chien, C.-F.: J. Intell. Manuf. **22**(3), 447–457 (2011)
161. Lee, C.-Y., Chien, C.-F.: OR Spectr. **36**(3), 761–797 (2014)
162. Chien, C.-F., Hsu, C.-Y.: J. Intell. Manuf. **22**(3), 399–412 (2011)
163. Chien, C.-F., Chen, J.-H., Wei, C.-C. (2011) **18**(4), 333–349
164. Chien, C.-F., Kim, K.H., Liu, B., Gen, M.: J. Intell. Manufact. 1–3 (2012)
165. Velásquez-Bermúdez, J.M., Khakifirooz, M., Fathi, M.: Large Scale Optimization in Supply Chains and Smart Manufacturing: Theory and Applications, vol. 149. Springer Optimization and Its Applications (2019)
166. Chien, C.-F., Peng, J.-T., Yu, H.-C.: Comput. Ind. Eng. **99**, 448–457 (2016)
167. Chien, C.-F., Chen, Y.-J., Hsu, C.-Y., Yeh, Y.-H.: In: Proceedings of the Winter Simulation Conference Winter Simulation Conference, pp. 1898–1907 (2011)
168. Chen, Y.-J., Hsu, C.-Y., Chien, C.-F.: In: Proceedings of 12th Asia Pacific Industrial Engineering & Management Systems Conference (APIEMS 2011) (2011)
169. Hsu, C.-Y., Chien, C.-F., Lai, Y.-C.: Main branch decision tree algorithm for yield enhancement with class imbalance. In: Intelligent Decision Technologies, pp. 235–244. Springer (2012)
170. Chien, C.-F., Chang, K.-H., Wang, W.-C.: J. Intell. Manuf. **25**(5), 961–972 (2014)
171. Chien, C.-F., Chen, Y.-J., Hsu, C.-Y.: Comput. Oper. Res. **53**, 309–318 (2015)
172. Chu, P.-C., Chien, C.-F., Chen, C.-C.: Int. J. Ind. Eng. **23**(5) (2016)
173. Chien, C.-F., Chen, Y.-J., Wu, J.-Z.: In: Winter Simulation Conference (WSC), 2016, pp. 2512–2522. IEEE (2016)
174. Chien, C.-F., Hsu, C.-Y., Chen, P.-N.: Flex. Serv. Manuf. J. **25**(3), 367–388 (2013)
175. Liu, C.-W., Chien, C.-F.: Eng. Appl. Artif. Intell. **26**(5–6), 1479–1486 (2013)
176. Liao, C.-S., Hsieh, T.-J., Huang, Y.-S., Chien, C.-F.: IEEE Trans. Autom. Sci. Eng. **11**(3), 953–960 (2014)
177. Hsieh, T.-J., Liao, C.-S., Huang, Y.-S., Chien, C.-F.: In: 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 869–874. IEEE (2012)
178. Chien, C.-F., Hsu, S.-C., Chen, Y.-J.: Int. J. Prod. Res. **51**(8), 2324–2338 (2013)
179. Chen, Y.-J., Lin, T.-H., Chang, K.-H., Chien, C.-F.: J. Ind. Prod. Eng. **30**(8), 510–517 (2013)
180. Chien, C.-F., Liu, C.-W., Chuang, S.-C.: Int. J. Prod. Res. **55**(17), 5095–5107 (2017)
181. Chien, C.-F., Diaz, A.C., Lan, Y.-B.: Int. J. Comput. Intell. Syst. **7**(sup2), 52–65 (2014)
182. Chien, C.-F., Chen, Y.-J.: In: 2016 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), pp. 1–4. IEEE (2016)
183. Chen, L.-F., Chien, C.-F.: Flex. Serv. Manuf. J. **23**(3), 263–289 (2011)
184. Chien, C.-F., Hu, C.-H.: Factor analysis system and analysis method thereof, US Patent 8,200,528 (2012)
185. Tamer, M., van derLans, M., Sadeghian, H.: In: Metrology, Inspection, and Process Control for Microlithography XXXII, vol. 10585, p. 105850O. International Society for Optics and Photonics (2018)
186. Park, J., Shin, C., Kim, M., Kim, J., Park, J., Kim, J., Jun, C., Yim, Y., Lee, J.: J. Micro/Nanolithography MEMS MOEMS **13**(4), 041409 (2014)
187. Kuo, H.-F., Faricha, A.: IEEE Access **4**, 7479–7486 (2016)
188. He, F., Zhang, Z.: RSC Adv. **5**(126), 103901–103906 (2015)
189. Xie, Q., Venkatachalam, P., Lee, J., Chen, Z., Zafar, K.: In: Metrology, Inspection, and Process Control for Microlithography XXXI, vol. 10145, p. 101452W. International Society for Optics and Photonics (2017)
190. Hong, T.-Y., Chien, C.-F., Wang, H.-K., Guo, H.-Z.: Comput. Ind. Eng. **125**, 200–211 (2018)
191. Jamrus, T., Chien, C.-F., Gen, M., Sethanan, K.: IEEE Trans. Semicond. Manuf. **31**(1), 32–41 (2018)
192. Chien, C.-F., Huynh, N.-T.: IEEE Trans. Semicond. Manuf. **31**(1), 76–86 (2018)

193. Wu, J.-Z., Hao, X.-C., Chien, C.-F., Gen, M.: J. Intell. Manuf. **23**(6), 2255–2270 (2012)
194. Hao, X., Lin, L., Gen, M., Chien, C.-F.: In: 2014 IEEE International Conference on Automation Science and Engineering (CASE), pp. 131–136. IEEE (2014)
195. Li, L., Min, Z.: Int. J. Adv. Manuf. Technol. **84**(1–4), 315–325 (2016)
196. Zhang, J., Wang, X.: Int. J. Prod. Res. **54**(23), 7043–7059 (2016)
197. Wang, S., Wang, L.: Knowl.-Based Syst. **84**, 1–9 (2015)
198. Kim, J., Chung, S.Y., Yoon, H.J.: Proc. Inst. Mech. Eng. Part B: J. Eng. Manuf. **228**(12), 1701–1712 (2014)
199. Cao, Z., Lin, C., Zhou, M., Huang, R.: IEEE Trans. Autom. Sci. Eng. (99), 1–13 (2018)
200. Waschneck, B., Reichstaller, A., Belzner, L., Altenmüller, T., Bauernhansl, T., Knapp, A., Kyek, A.: In 2018 29th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), pp. 301–306. IEEE (2018)
201. Cao, Z., Lin, C., Zhou, M., Huang, R.: In: 2017 13th IEEE Conference on Automation Science and Engineering (CASE), pp. 1040–1045. IEEE (2017)
202. Waschneck, B., Reichstaller, A., Belzner, L., Altenmüller, T., Bauernhansl, T., Knapp, A., Kyek, A.: Procedia CIRP **72**, 1264–1269 (2018)
203. Ma, Y., Qiao, F., Lu, J.: In: 2016 IEEE International Conference on Automation Science and Engineering (CASE), pp. 1394–1399. IEEE (2016)
204. Chen, Y.-T., Wu, C.-H., Tien, Y.-J., Yu, C.-J.: Int. J. Ind. Eng. **23**(5) (2016)
205. Hsu, C.-Y., Chien, C.-F., Chen, P.-N.: J. Chin. Inst. Ind. Eng. **29**(5), 303–313 (2012)
206. Tan, F., Pan, T., Li, Z., Chen, S.: IEEE Trans. Ind. Inform. **11**(6), 1435–1444 (2015)
207. Wang, Y., Zheng, Y., Gu, X.-G., Huang, L.: In: 2015 International Conference on Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII), pp. 152–155. IEEE (2015)
208. Wan, L., Pan, T.: In: 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pp. 394–397. IEEE (2015)
209. Bian, J., Pan, T.: In: 2014 11th World Congress on Intelligent Control and Automation (WCICA), pp. 4356–4360. IEEE (2014)
210. Wan, J., McLoone, S.: IEEE Trans. Semicond. Manuf. **31**(1), 12–21 (2018)
211. Korabi, T.E., Graton, G., Ouladsine, M., Pinaton, J., et al.: In: 2018 29th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), pp. 340–345. IEEE (2018)
212. Han, K., Wang, K.: J. Manuf. Syst. **32**(2), 372–381 (2013)
213. Liu, K., Chen, Y., Zhang, X.: IFAC-PapersOnLine **50**(1), 8097–8102 (2017)
214. Jou, B.-Y., Chan, Y.-T., Tseng, S.-T.: IEEE Trans. Semicond. Manuf. **25**(4), 614–622 (2012)
215. Minakata, T., Tanamura, M., Mitamura, Y., Imashiro, M., Horiguchi, A., Sugimoto, A., Yamashita, M., Ujiiye, K., Sunahiro, S., Yada, Y., et al.: In: SID Symposium Digest of Technical Papers, vol. 46-1, pp. 219–222. Wiley Online Library (2015)
216. Gong, Q.-S., Lee, M.-S., Yang, G.-K., Pan, C.-C.: In: 2015 International Conference on Control, Automation and Robotics (ICCAR), pp. 177–181. IEEE (2015)
217. Park, C.-S., Kim, J.S., Park, S.-H., Yun, J.-J., Baek, J.-G.: Adv. Sci. Lett. **14**(1), 458–463 (2012)
218. Kao, C.-A., Cheng, F.-T., Wu, W.-M., Kong, F.-W., Huang, H.-H.: IEEE Trans. Semicond. Manuf. **26**(1), 69–81 (2013)
219. Wan, L., Tan, F., Pan, T.-H.: Control. Theory Appl. **1**, 012 (2016)
220. Jebri, M., El Adel, E., Graton, G., Ouladsine, M., Pinaton, J.: IFAC-PapersOnLine **50**(1), 6154–6159 (2017)
221. Jen, C.H.: Adv. Materi. Res. (Trans Tech Publ) **630** 235–240 (2013)
222. Huang, S.-J., Chen, H.-Y.: Int. J. Control. Autom. Syst. **12**(2), 422–430 (2014)
223. Liu, J.P., Beyca, O.F., Rao, P.K., Kong, Z.J., Bukkapatnam, S.T.: IEEE Trans. Autom. Sci. Eng. **14**(1), 208–221 (2017)
224. Yang, H.-C., Tieng, H., Cheng, F.-T.: J. Chin. Inst. Eng. **39**(2), 221–235 (2016)
225. Lee, H., Kim, Y., Kim, C.O.: IEEE Trans. Semicond. Manuf. **30**(1), 23–31 (2017)
226. Pan, T.-H., Yang, Y.-L.: Control. Decis. **11**, 026 (2014)
227. Chen, X., Tomizuka, M.: Int. J. Adapt. Control Signal Process. **29**(11), 1459–1474 (2015)

228. Lynn, S.A., MacGearailt, N., Ringwood, J.V.: In: 2012 IEEE International Conference on Control Applications (CCA), pp. 1658–1663. IEEE (2012)
229. Vock, S., Escalona, O., Turner, C.: J. Electron. Test. **31**(1), 107–117 (2015)
230. Eleffendi, M.A., Johnson, C.M.: IEEE Trans. Power Electron. **32**(9), 7187–7198 (2017)
231. Chen, P.-C., Chang, M., Lai, W.-C., Gabayno, J.L.: J. Chin. Inst. Eng. **39**(4), 508–512 (2016)
232. Yoneda, Y., Nakamura, K.: FUJITSU Sci. Tech. J. **49**(1), 138–144 (2013)
233. Tai, Y., Pearn, W.: IEEE Trans. Semicond. Manufact. **28**(3), 424–430 (2015)
234. Liebens, M., Slabbekoorn, J., Miller, A., Beyne, E., Stoerring, M., Hiebert, S., Cross, A.: In: 2018 29th Annual SEMI Advanced Semiconductor Manufacturing Conference, pp. 10–17 IEEE (2018)
235. Park, S.H., Park, C.-S., Kim, J.S., Kim, S.-S., Baek, J.-G., An, D.: In: 2013 IEEE International Congress on Big Data (BigData Congress), pp. 363–368. IEEE (2013)
236. Wu, L., Wang, X., Li, W.: In: 2018 China Semiconductor Technology International Conference (CSTIC), pp. 1–4. IEEE (2018)
237. denBoef, A.J.: Surf. Topogr.: Metrol. Prop. **4**(2), 023001 (2016)
238. Hsu, C.-Y., Wu, J.-Z.: Int. J. Ind. Eng. **23**(5) (2016)
239. Tang, H., Shearer, J.C., Cheong, L.L., Saulnier, N.A., Sieg, S.A., Petrillo, K., Metz, A., Arnold, J.C.: J. Photopolym. Sci. Technol. **28**(1), 13–16 (2015)
240. Boumerzoug, M., Promreuk, S.: In: 2014 25th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), pp. 186–189. IEEE (2014)
241. Rizquez, M., Roussy, A., Pompier, D., Pinaton, J., Pasquet, J.: In: 2016 International Symposium on Semiconductor Manufacturing (ISSM), pp. 1–4. IEEE (2016)
242. Lee, H.K., Baek, K.H., Shin, K.: Jpn. J. Appl. Phys. **56**(6), 066502 (2017)
243. Kumar, P., Rosenbluth, A.E., Pusuluri, R.M., Viswanathan, R., Srinivasan, B., Mohapatra, N.R.: J. Micro/Nanolithography MEMS MOEMS **17**(2), 023503 (2018)
244. Susto, G.A., Pampuri, S., Schirru, A., Beghi, A., De Nicolao, G.: Comput. Oper. Res. **53**, 328–337 (2015)
245. Chen, W., Wang, Z., Chan, F.T.: Eur. J. Oper. Res. **261**(3), 929–940 (2017)
246. Sakai, M., Nishi, T.: Adv. Mech. Eng. **9**(4), 1687814017693217 (2017)
247. Kim, H.-J., Lee, J.-H., Lee, T.-E.: IEEE Trans. Autom. Sci. Eng. **12**(2), 690–700 (2015)
248. Nishi, T., Matsumoto, I.: IEEE Trans. Autom. Sci. Eng. **12**(1), 281–294 (2015)
249. Yang, T., Shen, Y.-A., Cho, C., Lin, Y.-R.: Eur. J. Ind. Eng. **6**(3), 281–300 (2012)
250. Manupati, V., Revanth, A., Srikanth, K., Maheedhar, A., Reddy, M.S.: Real-time rule-based scheduling system for integrated delivery in a semiconductor manufacturing using evolutionary algorithm-based simulation approach. In: Artificial Intelligence and Evolutionary Computations in Engineering Systems, pp. 981–989. Springer (2016)
251. Amaral, A.R.: Int. J. Prod. Res. 1–14 (2018)
252. Cong, P., Zhang, J., Wei, Q.: Chin. J. Mech. Eng. **30**(3), 663–675 (2017)
253. Paksoy, T., Karaoğlan, İ., Gökçen, H., Pardalos, P.M., Torğul, B.: J. Econ. Bibliogr. **3**(1S), 1–20 (2016)
254. Fang, C., Liu, X., Pardalos, P.M., Pei, J.: Int. J. Adv. Manuf. Technol. **83**(5–8), 689–710 (2016)
255. Liu, X., Pei, J., Liu, L., Cheng, H., Zhou, M., Pardalos, P.M.: Life cycle assessment in an IoT environment. In: Optimization and Management in Manufacturing Engineering, pp. 209–246. Springer (2017)
256. Frederix, F.: In: Proceedings of the Conference on Integration in Manufacturing, Galway, Ireland, pp. 107–116 (1996)
257. Callarman, T., Fowler, J., Gel, E., Pfund, M., Shunk, D.: Creating a research agenda framework for semiconductor supply network integration. In: Evolution of Supply Chain Management, pp. 161–187. Springer (2004)
258. Sun, Y., Feller, A., Shunk, D., Fowler, J., Callarman, T., Duarte, B.: In: IEEE International Conference on Automation Science and Engineering, 2007, CASE 2007, pp. 106–110. IEEE (2007)

259. Fordyce, K., Milne, R.J., Wang, C.-T., Zisgen, H.: Int. J. Ind. Eng.: Theory Appl. Pract. **22**(5), 575–600 (2015)
260. Fordyce, K., Milne, R.J., Wang, C.-T., Zisgen, H.: Int. J. Ind. Eng.: Theory Appl. Pract. **22**(5), 601–617 (2015)
261. Bahinipati, B.K., Deshmukh, S.: Logist. Res. **4**(1–2), 19–38 (2012)
262. Mönch, L., Uzsoy, R., Fowler, J.W.: Int. J. Prod. Res. 1–22 (2017)
263. Liu, X.P.J.L.L.C.H.Z.M.P.P. (2018) Int. J. Prod. Res. 1–19 (2018)
264. Uzsoy, R., Fowler, J.W., Mönch, L.: Int. J. Prod. Res. 1–19 (2018)
265. Teimoury, E., Fathi, M.: Int. J. Prod. Res. **51**(18), 5576–5596 (2013)
266. Teimoury, E., Modarres, M., Khondabi, I., Fathi, M.: Int. J. Adv. Manuf. Technol. **63**(1–4), 359–371 (2012)
267. Centobelli, P., Cerchione, R., Esposito, E.: Transp. Res. Part D: Transp. Environ. **53**, 454–470 (2017)
268. Kwak, M., Behdad, S., Zhao, Y., Kim, H., Thurston, D.: J. Mech. Des. **133**(10), 101003 (2011)
269. Sabbaghi, M., Behdad, S., Zhuang, J.: Int. J. Prod. Econ. **182**, 545–563 (2016)
270. Xue, M., Yan, G., Li, J., Xu, Z.: Environ. Sci. Technol. **46**(19), 10556–10563 (2012)
271. Pardalos, P.M.: Smart/green manufacturing: data enabled decision making and optimization applications-presentation (2017). http://www.ise.ufl.edu/cao/
272. Centobelli, P., Cerchione, R., Esposito, E.: Sustainability **9**(3), 361 (2017)

Chapter 12
Feedback Control Algorithms for the Dissipation of Traffic Waves with Autonomous Vehicles

Check for updates

Maria Laura Delle Monache, Thibault Liard, Anaïs Rat, Raphael Stern, Rahul Bhadani, Benjamin Seibold, Jonathan Sprinkle, Daniel B. Work and Benedetto Piccoli

Abstract This article considers the problem of traffic control in which an autonomous vehicle is used to regulate human-piloted traffic to dissipate stop-and-go traffic waves. We first investigated the controllability of well-known microscopic traffic flow models, namely, (i) the Bando model (also known as the optimal velocity model), (ii) the follow-the-leader model, and (iii) a combined optimal velocity

M. L. Delle Monache · T. Liard
University of Grenoble Alpes, Inria, CNRS, Grenoble INP, GIPSA-Lab,
38000 Grenoble, France
e-mail: ml.dellemonache@inria.fr

T. Liard
e-mail: thibault.liard@inria.fr

A. Rat · B. Piccoli (✉)
Department of Mathematics, University of Rutgers, Camden, NJ, USA
e-mail: piccoli@camden.rutgers.edu

A. Rat
e-mail: anais.rat@gmail.com

R. Stern · D. B. Work
Department of Civil and Environmental Engineering, Institute for Software
Integrated Systems, Vanderbilt University, 2301 Vanderbilt Place, Nashville,
TN 37235-1826, USA
e-mail: raphael.stern@vanderbilt.edu

D. B. Work
e-mail: dan.work@vanderbilt.edu

R. Bhadani · J. Sprinkle
Department of Electrical and Computer Engineering, University of Arizona,
1230 E. Speedway Blvd., Tucson, AZ 85721-0104, USA
e-mail: rahulbhadani@catworks.arizona.edu

J. Sprinkle
e-mail: sprinkle@ece.arizona.edu

B. Seibold
Department of Mathematics, Temple University, 1805 N. Broad Street,
Philadelphia, PA 19122, USA
e-mail: seibold@temple.edu

follow-the-leader model. Based on the controllability results, we proposed three control strategies for an autonomous vehicle to stabilize the other, human-piloted traffics. We subsequently simulate the control effects on the microscopic models of human drivers in numerical experiments to quantify the potential benefits of the controllers. Based on the simulations, finally, we conduct a field experiment with 22 human drivers and a fully autonomous-capable vehicle, to assess the feasibility of autonomous vehicle-based traffic control on real human-piloted traffic. We show that both in simulation and in the field test that an autonomous vehicle is able to dampen waves generated by 22 cars, and that as a consequence, the total fuel consumption of all vehicles is reduced by up to 20%.

12.1 Introduction

Currently, the vehicular transportation system is undergoing a major transition from vehicles in which humans are responsible for all driving tasks, to one in which automation is responsible for all driving tasks. The transition is defined [49] in terms of various levels of automation. The levels range from *level one autonomous vehicles* (AVs) available today that provide the driver with minor technological assistance (e.g., stability control or lane correction assist), to *level five* AVs which operate autonomously in all scenarios and in which humans cannot intervene. As the penetration rate of vehicles at each level of automation increases and shifts up the scale, new opportunities are arising to use automated vehicles to begin controlling the overall traffic flow.

A paradigm of traffic control in which some automated vehicles are also acting as traffic control devices is beginning to emerge. In particular, the works [10, 16, 26, 57, 58], explore the possibility of adaptive cruise controlled vehicles (e.g., level one automation) to influence traffic flow, for example, by smoothing the flow and/or increasing the flow rate. On the extreme end, when the adaptive cruise controlled vehicles are also endowed with communication capabilities between vehicles (referred to as *Cooperative Adaptive Cruise Control* (CACC) systems), small headways can be achieved and substantial increases in freeway throughput can be obtained [3, 5, 9, 12, 23, 29, 44, 50, 54]. On the experimental side, field experiments with commercial adaptive cruise control vehicles illustrate current technology may, in fact, *amplify* traffic oscillations [33], while correctly designed CACC systems dissipate these oscillations [14, 33]. An experiment to harmonize speeds on a US freeway was recently reported [31, 32].

The use of vehicles as traffic controllers may also integrate with more classical traffic control infrastructure, such as ramp meters and variable speed limit systems [15, 19, 20, 35, 37, 38, 43, 51], or systems which combine the two strategies [17, 18, 30, 39]. One limitation of the infrastructure-based solutions is their limited spatial resolution, as well as the need for driver compliance in the case of speed advisory based systems. The interest to use automated vehicles for traffic control is partly motivated by these limitations.

Considering the new direction of traffic control in which automated vehicles act as actuators, the main contribution of this article is to propose and assess control algorithms designed to dissipate *stop-and-go* traffic waves with an autonomous vehicle. Stop- and-go traffic waves are present on freeways and have many triggering events, such as lane changes. Strikingly, in the seminal experiments of Sugiyama et al. [52] and Tadaki et al. [55], human driving behavior alone was shown to be sufficient to trigger stop-and-go waves. The experiments were conducted on a single-lane circular track with real human drivers, and the uniform flow at the experiment start, quickly breaks down into a persistent stop-and-go wave that travels against the flow of traffic.

Motivated by this experiment, we design control algorithms to be implemented on an autonomous vehicle with the goal of dissipating stop-and-go waves caused by human driving behavior. We proceed as follows. First, we model the vehicular traffic at the microscopic scale using one or a combination of two well-known microscopic models, namely, (i) the Bando or *optimal velocity* (OV) model [1] and the (ii) *follow the leader* (FTL) one [42, 47, 48]. With the models defined, we show that traffic described by a linearization of the FTL model is not controllable via a single AV, implying the nonlinear FTL model is not linearly controllable. This result prevents the use of simple linear controllers to stabilize the traffic around the uniform speed equilibrium traffic state. On the other hand, we show that traffic described by the optimal velocity model is locally controllable by an autonomous vehicle. We show that for driving dynamics, that include both the optimal velocity and follow the leader terms, the resulting model is also locally controllable.[1]

We then proceed to design three different controls in which the AV is used to dampen stop-and-go waves. The first two are based on Lyapunov functions and only require measurements of the AV speed and the speed of nearby vehicles, while the third is an PID-type control in which the AV is controlled using only measurements of its own speed over several proceeding timesteps.

We first assess the effectiveness of the wave dampening controllers though numerical experiments. In addition to showing a reduction in the wave strength, we also quantify the benefits in terms of a reduction of total fuel consumed by all vehicles in the simulation. Based on the positive performance in simulation, we proceed to field validate one control algorithm with a dive-by-wire autonomous- capable vehicle on a track with 22 vehicles driven by humans. The experiment shows that stop-and-go waves can be dampened, and the projected reduction of fuel consumption from the simulations (approximately 20%) is confirmed via real-time fuel consumption loggers installed on the experimental vehicles.

The remainder of the article is organized as follows. In Section 12.2, we review the main microscopic traffic flow models investigated in this work. Section 12.3 establishes the main controllability results of the models, while Section 12.4 describes the design and testing the wave dissipating controllers in simulation. In Section 12.5, we test one of the wave dampening controllers in a field experiment with real human drivers and an autonomous vehicle. Limitations and future directions are explored in Section 12.6.

---

[1]Established for $n \leqslant 9$ vehicles and conjectured for $n > 9$.

12.2   Microscopic Traffic Models

Traffic models are usually defined in categories, depending on the scale at which they represent vehicular traffic, including microscopic, mesoscopic, macroscopic, and cellular. For a review of models at various scales see [2, 13, 21, 41]. Moreover, some approaches are based on model-agnostic simulation tools, such as deep learning and neural networks [7, 22, 25, 60].

In this paper, we focus on microscopic models. Microscopic models are suitable for in-silico verification before experimental testing since they describe human driving behavior at the individual vehicle level. One such model is the combined Bando and follow the leader model, which can be formulated as $\dot{x}_i = v_i$, $\dot{v}_i = f(x_{i+1}, x_i, v_{i+1}, v_i)$, where x_i is the position of ith car, v_i its velocity, and $i + 1$ is the index of the car ahead.

More precisely, for what concerns the follow-the-leader model, it was introduced in [42, 47, 48] and it assumes that the acceleration of a vehicle is given by the neighboring vehicles. The main influence comes from the next vehicle, whose index is $i + 1$, that is also called leading vehicle. The main dynamics described by this model is given by

$$\begin{cases} \dot{x}_i = v_i \\ \dot{v}_i = C\, \dfrac{v_{i+1} - v_i}{x_{i+1} - x_i} \qquad 1 \leqslant i \leqslant N \end{cases} \tag{12.1}$$

where C is a constant with appropriate dimension and, for simplicity, from now on we set $C = 1$. This model has the following properties:

- The acceleration depends on the relative velocity $\Delta v = v_{i+1} - v_i$.
- The velocity $v_i(t)$ of the vehicle depends on the velocity of the vehicle in front such that the distance from the vehicle in front is safe.

A drawback of these models is that the acceleration is zero when the relative velocity is zero independently of the headway $d \equiv x_{i+1} - x_i$. That is, extremely small headways are allowed even when traveling with extremely high speed, [36]. A model that fixed this problem is the optimal velocity model (introduced by [1]) that describes the adaptation of the actual speed to the optimal velocity $V(\cdot)$ which stands for the desired speed defined by

$$V(x) = v_{\max}\, \frac{\tanh(x - l_v - d_s) + \tanh(l_v + d_s)}{1 + \tanh(l_v + d_s)}, \tag{12.2}$$

where l_v is the length of cars and $d_s > 0$ is the safe distance between cars. The optimal velocity has the property that it tends to zero for small headways and it achieves the maximum value for large headways. The full model reads:

$$\begin{cases} \dot{x}_i = v_i \\ \dot{v}_i = V(x_{i+1} - x_i) - v_i. \qquad 1 \leqslant i \leqslant N \end{cases} \tag{12.3}$$

In this model, a driver controls acceleration or deceleration according to the difference between the optimal velocity and his own velocity. The equilibrium point for this model is achieved when all cars are at constant speed and have the same headway [27]. For this model, it is possible to derive a feedback law such that the controlled traffic system is stable [34].

12.3 Controllability Results for Microscopic Models

In this section, we provide theoretical results about controllability for the microscopic models introduced in Section 12.2. More precisely, we will focus on a ring-road setting, which reproduces the situation of the celebrated Sugiyama experiment [53] with a single AV, which can be controlled, and investigate the controllability of the corresponding control system.

Let us first recall some basic facts about control systems, referring the reader to the books [4] and [8] for details. A control systems is a dynamical system written as

$$\dot{y} = f(y, u) \tag{12.4}$$

where $y \in \mathbb{R}^n$ represents the state of the system and $u \in U \subset \mathbb{R}^m$ represents the control vector, i.e., the parameters which can be chosen by an external agent. In our setting, y represent the state of the traffic model (e.g., position and velocity of cars) and u the acceleration or speed of the AV which can be controlled.

A system (12.4) is said to be controllable if for every states y_1, y_2, there exists $T > 0$ and a control function $\bar{u} : [0, T] \to U$ such that the solution to the Cauchy problem $\dot{y} = f(t, \bar{u}(t))$, $y(0) = y_1$ satisfies $y(T) = y_2$. In other words, we can steer the system from y_1 to y_2 in time $T > 0$ with a suitable control $\bar{u}$.

Similarly, a system is said to be locally controllable at $\bar{y}$ if for every $\delta > 0$ sufficiently small, there exists $T > 0$ such that we can steer the system from $\bar{y}$ to any y with $|y - \bar{y}| < \delta$ in time T. In other words, we can reach sufficiently close states in uniformly bounded time.

An important tool to investigate controllability (and local ones) is linearization. Consider a control system (12.4) such that U contains a neighborhood of 0, and an equilibrium point $(\bar{y}, 0)$, i.e., $f(\bar{y}, 0) = 0$, we can consider the linearized system at $\bar{y}$:

$$\dot{z} = A \cdot z + B \cdot u, \tag{12.5}$$

where $A = D_y f(\bar{y}, 0)$ (the Jacobian matrix of f w.r.t. y computed at $(\bar{y}, 0)$) and $B = D_u f(\bar{y}, 0)$ (the Jacobian matrix of f with respect to u computed at $(\bar{y}, 0)$). For linear systems, there is a simple criterion for controllability. First, given the linear control system (12.5), define the *Kalman controllability matrix*:

$$K(A, B) = [B, A \cdot B, \ldots, A^{n-1} \cdot B]. \tag{12.6}$$

Since A is an $n \times n$ matrix and B is a $n \times m$ matrix, then $K(A, B)$ is an $n \times n \cdot m$ matrix. We can now state the following (see [4, Theorem 3.6.2]):

Theorem 12.1 *(Kalman controllability theorem) The system* (12.5) *is controllable if and only if the matrix $K(A, B)$ has full rank (i.e., equal to n).*

The fact that we can limit ourselves to the exponent $n - 1$ in the definition of $K(A, B)$ follows from Cayley–Hamilton Theorem, which ensures that A, which is an $n \times n$ matrix, is root of its characteristic polynomial. Since the characteristic polynomial has degree n then there exists α_i such that $A^n = \sum_{i=0}^{n-1} \alpha_i A^i$.

The local controllability of a system can be investigated by looking at its linearization. More precisely we have the following (see [4, Theorem 3.7.1]):

Theorem 12.2 *Consider a control system* (12.4) *and its linearization* (12.5) *around the equilibrium $(\bar{y}, 0)$. If* (12.5) *is controllable then* (12.4) *is locally controllable at $(\bar{y}, 0)$.*

The converse of this Theorem is not true: a nonlinear control system may be controllable even if its linearization fails to be controllable. However, the controllability of the linearization ensures the existence of linear stabilizing feedbacks, i.e., control laws of the type $u(y) = K \cdot y$ that renders the system (12.4) locally asymptotically stable at $(\bar{y}, 0)$, see [4, Theorem 4.2.3].

We are now ready to define our control system for traffic control on a ring-road via one AV, see Figure 12.1. Let $(a, b) \in (\mathbb{R}_+)^2 \backslash \{(0, 0)\}$ and consider the control system of n vehicles along a ring-road of length L described by the FTL-Bando model:

$$\begin{cases} \dot{x}_i = v_i, \quad 1 \leqslant i \leqslant n, \\ \dot{v}_i = a \, \dfrac{v_{i+1} - v_i}{(x_{i+1} - x_i)^2} + b \left[V(x_{i+1} - x_i) - v_i \right], \quad 1 \leqslant i \leqslant n-1, \quad (12.7) \\ \dot{v}_n = u, \end{cases}$$

Fig. 12.1 Sketch of n vehicles on a trajectory along a ring of length L
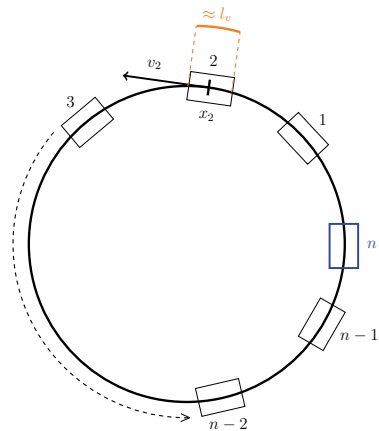
where x_i is the position of the ith vehicle, v_i its velocity and $V(\cdot)$ stands for a desired speed which is defined in Section 12.2. Notice that we can control only the acceleration u of the nth vehicle, which represents the AV.

Remark 12.1 If $b = 0$, the nonlinear control system (12.7) is the Follow-The-Leader model (FTL). If $a = 0$, the system (12.7) is the optimal velocity model.

Our aim is to steer the system, (12.7) to a speed equilibrium state, i.e., a state so that all vehicles have the same velocity $\bar{v}$. If $b \neq 0$, i.e., if there is a non vanishing optimal velocity term, a speed equilibrium corresponds to all vehicles which are uncontrolled (i.e., all vehicles but the nth one) having the same headway distance d, while the AV may have a different headway. Moreover, to be in speed equilibrium we must have $V(d) = \bar{v}$, thus d is fixed once $\bar{v}$ is fixed. On the other hand if $b = 0$ then all states with same speed are an equilibrium with any set of headways. We summarize this analysis in the following Lemma:

Lemma 12.1 *Consider the control system* (12.7) *and a fixed speed* $\bar{v}$.
If $b \neq 0$ *then the following holds. Let d be such that $V(d) = \bar{v}$. If $d \leqslant \frac{L}{n}$ then the only speed equilibrium is given by $x(t)$ such that $x_{i+1}(t) - x_i(t) \equiv d$, $i = 1, \ldots, n-1$ and $\dot{x}_i(t) \equiv \bar{v}$, $i = 1, \ldots, n$. Otherwise, there exists no speed equilibrium. If $b = 0$ then every $x(t)$ such that $\dot{x}_i(t) \equiv \bar{v}$, $i = 1, \ldots, n$, is an equilibrium.*

Now, fix $(d, \bar{v}) \in \mathbb{R}_+^2$, with $V(d) = \bar{v}$ if $b \neq 0$, then we rewrite (12.7) as

$$\dot{Y} = f(Y, u). \tag{12.8}$$

where $Y = (y_i)_{i=1,\ldots,2n-1} = (x_2 - x_1 - d, \ldots, x_n - x_{n-1} - d, v_2 - v_1, \ldots, v_n - v_{n-1}, v_n - \bar{v})$ and f is defined by

$$f(Y, u) = \begin{pmatrix} y_n \\ \vdots \\ y_{2n-2} \\ a\left[\dfrac{y_{n+1}}{(y_2+d)^2} - \dfrac{y_n}{(y_1+d)^2}\right] + b\left[V(y_2+d) - V(y_1+d) - y_n\right] \\ \vdots \\ a\left[\dfrac{y_{2n-2}}{(y_{n-1}+d)^2} - \dfrac{y_{2n-3}}{(y_{n-2}+d)^2}\right] + b\left[V(y_{n-1}+d) - V(y_{n-2}+d) - y_{2n-3}\right] \\ u - a\,\dfrac{y_{2n-2}}{(y_{n-1}+d)^2} - b\left[V(y_{n-1}+d) - \left(y_{2n-1} - y_{2n-2} + \bar{v}\right)\right] \\ u \end{pmatrix} \tag{12.9}$$

Notice that the vector Y has $2n - 1$ component: $n - 1$ differences between the headway of the $n - 1$ uncontrolled vehicles and the equilibrium headway d, $n - 1$ velocity differences with leading vehicle for the uncontrolled vehicles and the difference of AV velocity with the equilibrium ones. Moreover we have the relation

$x_1 - x_n = L - \sum_{i=1}^{n-1} y_i + (n-1)d$ expressing the headway of the AV. We are interested in speed equilibria with equispaced cars, thus we notice that $Y^* = 0 \in \mathbb{R}^{2n-1}$ is the only equilibrium of (12.8) if $b \neq 0$ and $u = 0$. In other words $(Y^*, 0)$ is a speed equilibrium of the controlled system (12.8), i.e., $f(Y^*, 0) = 0$. Notice that for $d = \frac{L}{n}$, $V(d) = \bar{v}$, at the speed equilibrium $(0, 0)$, all vehicles are equispaced and drive at the same speed $v_i = \bar{v}$.

We are now ready to state the controllability results for the linearized systems at equilibria. First, we have the following:

Theorem 12.3 *Let $b = 0$, $a \neq 0$ and $(d, \bar{v}) \in \mathbb{R}_+^2$. The linearization of system (12.8) at the speed equilibrium $(Y^*, 0) \in \mathbb{R}^{2n}$ is not controllable.*

In other words, the linearization FTL model at speed equilibrium is not controllable. This does not prevent the nonlinear system to be controllable, but prevents the use of simple linear controls to drive the system to equilibrium.

Our next result is the following:

Theorem 12.4 *Let $a = 0$, $b \neq 0$, $(d, \bar{v}) \in \mathbb{R}_+^2$ with $V(d) = \bar{v}$. The linearization of system (12.8) at the speed equilibrium $(Y^*, 0) \in \mathbb{R}^{2n}$ is controllable. Therefore, the nonlinear control system (12.8) is locally controllable at $(Y^*, 0)$.*

Finally for the combined FTL-OV model we have the following:

Theorem 12.5 *We assume that $n \leqslant 9$. Let $(a, b) \neq 0$, $(d, \bar{v}) \in \mathbb{R}_+^2$ with $V(d) = \bar{v}$. If $a \neq (y^* + d)V'(y^* + d)$ and $b \neq 0$, the linearization of system (12.8) at the equilibrium point $(Y^*, 0) \in \mathbb{R}^{2n}$ is controllable. Therefore, the nonlinear control system (12.8) is locally controllable at $(Y^*, 0)$.*

Remark 12.2 Since the Optimal Velocity model is locally controllable, one can expect the same to be true for the combined FTL-OV model, except possibly some resonant value of the parameters. This is exactly what it proved in Theorem 12.5. The complexity of the system does not allow to deal with arbitrary large dimension (only for $n \leqslant 9$), but we conjecture that, for $n > 9$, the nonlinear control system (12.8) is still locally controllable at $(Y^*, 0)$.

The proofs of the Theorems are postponed to the Appendix.

12.4 Controls and Simulations

In this section, we describe control algorithms to stabilize a FTL-Bando model to a speed equilibrium and test them in silico, i.e., via simulations, on a setting reproducing the Sugiyama experiment [53].

We first define controls based on Lyapunov-type functionals, using the fact that the system is control affine, see [8, Definition 3.12], i.e., the control appear linearly with a vector coefficient depending on the state. Such techniques are usually referred

to as Jurdjevic–Quinn controls [24]. This analysis allows to understand the stability properties of the system and inspires the design of more practical controls.

We then define a Proportional–Integral–Derivative (PID)-type control, which is based on velocity measures of the AV over a fixed time horizon. The control includes saturation terms to avoid collisions and too large headways for the AV.

12.4.1   Lyapunov-Type Functions and Feedback Control

The system (12.8) is a control affine system (see [8, Definition 3.12]). More precisely, (12.8) can be written as

$$\dot{Y} = f_{a,b}(Y) + u f_1(Y),$$

with $f_{a,b}(Y)$ given by (12.9) with $u = 0$ and

$$f_1(Y) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 1 \end{pmatrix}.$$

Since we want to steer the system toward the equilibrium speed, we can start considering the Lyapunov-type function $V_1(y_{2n-1}) = \frac{(v_n - \bar{v})^2}{2} = \frac{y_{2n-1}^2}{2}$ associated to the equation $\dot{y}_{2n-1} = u$. Here, we aim at the controlled car to align its speed to the desired one and this may be sufficient to stabilize the system. The derivative of V along trajectories is given by $\nabla V_1 \cdot u = y_{2n-1}u$. Thus, we can simply choose

$$u_1 = -\alpha y_{2n-1} = -\alpha(v_n - \bar{v}), \tag{12.10}$$

where $\alpha > 0$ is chosen such that $-\alpha y_{2n-1} \in U$. Such control is smooth, vanishes at 0, and a good candidate to stabilize the system (12.8) to the equilibrium speed.

Let us now focus on last two components $\tilde{y} = (y_{2n-2}, y_{2n-1})$. We consider the following system of two equations:

$$\dot{\tilde{y}} = \begin{pmatrix} \dot{y}_{2n-2} \\ \dot{y}_{2n-1} \end{pmatrix} = \begin{pmatrix} u - a \frac{y_{2n-2}}{(y^*+d)^2} \\ u \end{pmatrix}, \tag{12.11}$$

and call $g = g(\tilde{y}, u)$ the function on the right-hand side. We introduce the Lyapunov-type functional associated to (12.11): $V_2(\tilde{y}) = \frac{y_{2n-1}^2}{2} + \frac{y_{2n-2}^2}{2}$. For every $\tilde{y} \in \mathbb{R}^2 \backslash \{(0,0)\}$, choosing $u = -\alpha \mathrm{sign}(y_{2n-1} + y_{2n-2})$ with $\alpha > 0$ and $\pm\alpha \in U$ we have, $\nabla V_2(\tilde{y}) \cdot g(\tilde{y}, u) = (y_{2n-2} + y_{2n-1})u - a \frac{y_{2n-2}^2}{(y^*+d)^2} < 0$. Thus, the function V_2 satisfies the

small control property associated to (12.11) (see [8, Definition 12.1]). The feedback control function

$$u_2 = -\alpha(y_{2n-1} + y_{2n-2}) = -\alpha\left(v_n - \frac{v_{n-1} + \bar{v}}{2}\right),\qquad(12.12)$$

where $\alpha > 0$ is chosen such that $-\alpha(y_{2n-1} + y_{2n-2}) \in U$, is smooth, vanishes at $0 \in \mathbb{R}^2$ and a good candidate to stabilize the control system (12.11) to the equilibrium speed.

12.4.2 PID Control

The idea behind this controller is that the autonomous vehicle may estimate the average speed of the vehicles in front, and then drive according to the average speed, safety permitting. An estimate of the average speed required by the controller is obtained by measuring the autonomous vehicle speed over a large enough time horizon. Note that this requires that there are several waves present, not just a single one, so that the past is informative of the future.

The controller determines a command velocity u following a standard proportional-integral control logic. In order for the controller to be efficient, it needs to be augmented with saturation: for small gaps, the autonomous vehicle should follow the lead vehicle speed to avoid dangerous situations, while for large gaps, the autonomous should catch up to the lead vehicle.

More precisely, this controller estimates the desired velocity, V_d, as a temporal average of the autonomous vehicle's own velocity over an interval. Letting $v_1^{AV}, \ldots, v_m^{AV}$ denote the autonomous vehicles velocities over the last m measurements, the desired velocity is computed as the temporal average $V_d = \frac{1}{m}\sum_{j=1}^{m} v_j^{AV}$. In practice, we choose m corresponding to a 38 s interval, which is approximately the time required to travel one lap around the ring.

The desired average velocity is then translated into a target velocity depending on the current gap between the autonomous vehicle and lead vehicle:

$$v^{\text{target}} = V_d + 1\tfrac{\text{m}}{\text{s}} \times \min(\max(\tfrac{\Delta x - 7\text{m}}{23\text{m}}, 0), 1),\qquad(12.13)$$

This allows the autonomous vehicle to drive faster than the average velocity and catch up to the lead vehicle, should it face a big gap, while at lower gaps, the target velocity reduces to the average V_d.

The commanded velocity is updated via

$$u_{j+1} = \beta_j(\alpha_j v_j^{\text{target}} + (1 - \alpha_j)v_j^{\text{lead}}) + (1 - \beta_j)u_j,\qquad(12.14)$$

where the subscript j denotes the time step. This rule (12.14) chooses the new commanded velocity as a weighted average of the prior commanded velocity, the

target velocity, and the lead vehicle's velocity. The weights α_j and β_j depend on the gap as follows:

$$\alpha_j = \min(\max(\frac{\Delta x - \Delta x^s}{\gamma}, 0), 1) \tag{12.15}$$

In (12.15), the distance Δx^s is a safety distance. We have $\alpha_j = 0$ if $\Delta x \leq \Delta x^s$ and $\alpha_j = 1$ if $\Delta x \geq \Delta x^s + \gamma$, meaning that for relatively short gaps, only the lead vehicle's velocity matters, while for relatively large gaps, only the target velocity is averaged with the commanded velocity. The parameter γ controls the rate at which α transitions from 0 to 1, and is set to $\gamma = 2$ m in the current implementation. This means that when the gap is short, the autonomous vehicle has the same speed as the lead vehicle, while when the gap is larger the autonomous speed tends toward the target vehicle, which allows the autonomous vehicle to reduce the gap with the lead vehicle. The parameter β_j determines how rapidly the controller adjusts to new situations (with more rapid adjustments occurring in more safety-critical situations). At its core, this is a PID controller, but with a saturation at small gaps (for safety purposes), and a saturation at large gaps (so that the autonomous vehicle closes gaps).

The safety distance is implemented as $\Delta x^s = \max(2 \text{ s} \times \Delta v, 4 \text{ m})$. The term $2 \text{ s} \times \Delta v$ represents the recommended safe following headway of 2 s, with a lower bound of 4 m.

12.4.3   Simulations

In this section, we demonstrate the capabilities of the control laws described in the previous sections via numerical simulations. The parameters used for the simulations are as follows. We consider $N = 22$ vehicles, with one autonomous vehicle and $N = 21$ human-driven vehicles. We consider that the human-driven vehicles follow the dynamics described in (12.7) with the following parameters $a = 0.5 \text{ m/s}^2$ and $b = 20 \text{ s}^{-1}$. The speed $v_{\max} = 9.75 \text{ m/s}$ and the vehicle length is chosen to be $l_v = 4.5$ m. Such parameters allow to fit the data from the experiment of [53], see Figure 12.2. In the figure, the red trajectory corresponds to the autonomous vehicle and the gray ones to human-driven vehicles. One can notice the appearance of strong stop and go waves that start at time 60 s and are propagated on the ring. Visually, they have the effect of "wrinkles" in the pattern.

For comparison, we can now see the effects of the different controls. For the autonomous vehicle, we will choose a different dynamics according to the control strategy that we are going to simulate. The simulation begins with all vehicles using the human driving dynamics described in (12.7) with no control. Control on the AV is activated after $t = 40$ s.

In Figure 12.3, we can see the effects of the PID control on vehicle trajectories and on the speed profile in Figure 12.4. Figure 12.3 shows that the AV leaves some extra headway to tame the effect of the stop-and-go waves. Such waves continue to reappear but the control is able to dissipate partly the effects. The velocity profiles, Figure 12.4, show a strong oscillation reduction for the AV in the time interval 60,
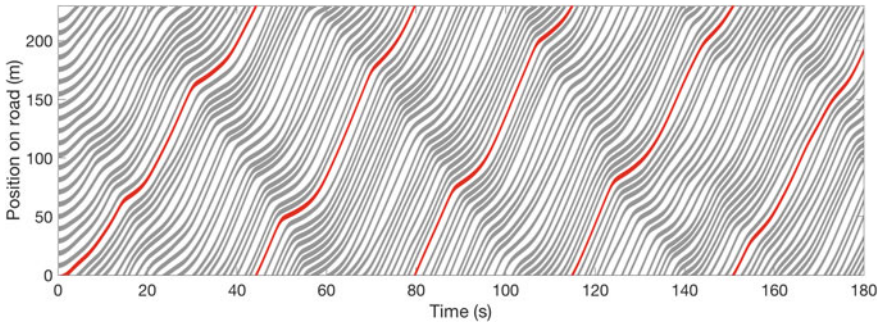
Fig. 12.2 Vehicle trajectories with no control. Note the red trajectory follows a single vehicle which acts identically to all gray trajectories
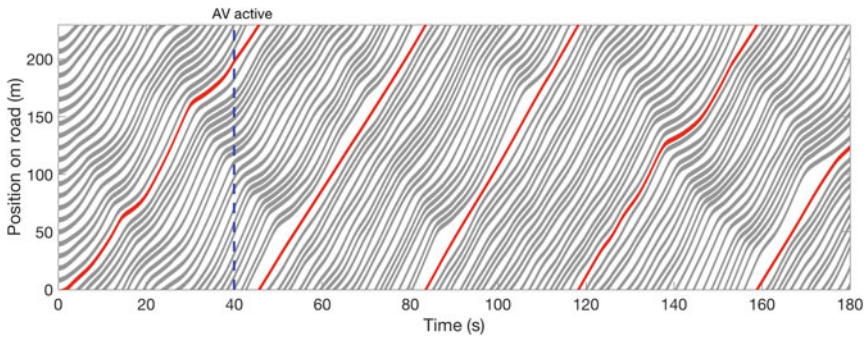

Fig. 12.3 Vehicle trajectories of uncontrolled vehicles (gray) when using the PID control (12.14) applied only to a single vehicle (red)
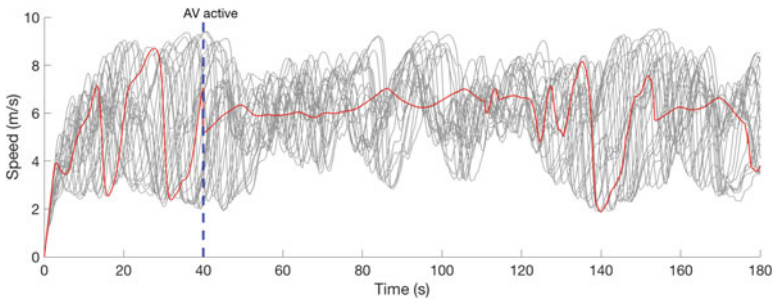

Fig. 12.4 Velocity profiles of uncontrolled vehicles (gray) when applying the PID control to a single vehicle (red) (12.14)

100, then oscillations affect the AV and finally are more under control after time $t = 160$.

Next, we show that stop-and-go waves can be dampened using the control laws (12.10) and (12.12). Since the autonomous vehicle used in the experiment runs with

velocity control, we will slightly modify our controls (12.10) and (12.12) as follows; let $u_j \in \mathbb{R}_+$ be defined by $u_j = u(t_j)$ with $t_j = j * \Delta t$, $j \in \mathbb{N}$ and $\Delta t > 0$.
First, combining (12.10) with (12.7), we have $\dot{v}_n = -(v_n - \bar{v})$ with $\bar{v} \in \mathbb{R}_+$. Thus, the target velocity is

$$v_{1,j+1}^{\text{target}} = (u_j - \bar{v}) \exp(-\Delta t) + \bar{v}. \tag{12.16}$$

Since $v_{n-1}(s) = v_{n-1}(t) := v_{n-1,j}$ for every $s \in [t_j, t_j + \Delta t)$, combining (12.12) with (12.7), the target velocity is

$$v_{2,j+1}^{\text{target}} = \left(u_j - \frac{v_{n-1,j} + \bar{v}}{2} \right) \exp(-\Delta t) + \frac{v_{n-1,j} + \bar{v}}{2}. \tag{12.17}$$

From now on, $\bar{v}$ is a function depending on time and at every time $t > 0$, $\bar{v}(t)$ is constructed as the minimum of the temporal average speeds of the lead vehicle and the autonomous vehicle over $(0, t)$. More precisely, the target velocity is

$$\bar{v}_j = \min \left(\frac{\sum_{i=1}^{j-1} v_i^{\text{lead}}}{j-1}, \frac{\sum_{i=1}^{j-1} u_i}{j-1} \right) \tag{12.18}$$

The theory described in Section 12.4.1 is still useful because, for j large enough so that one measures data over a complete run of the ring road, $v_j \approx c$ with $c > 0$ a constant. Adding the rule (12.14) described in Section 12.4.2, the commanded velocities for (12.16) and (12.17) are updated via

$$u_{j+1} = \beta_j(\alpha_j v_{k,j}^{\text{target}} + (1 - \alpha_j)v_j^{\text{lead}}) + (1 - \beta_j)u_j, \quad k \in \{1, 2\}. \tag{12.19}$$

where $v_{1,j}^{\text{target}}$ and $v_{2,j}^{\text{target}}$ are defined in (12.16) and (12.17), respectively, replacing $\bar{v}$ by $\bar{v}_j$ defined in (12.18). Above, α_j and β_j are constructed as explained in Section 12.4.2. In Figure 12.5 and in Figure 12.6, we use the commanded velocity defined in (12.19) with $v_{1,j}^{\text{target}}$ and $v_{2,j}^{\text{target}}$ respectively. Since the autonomous vehicle drives according to (12.18), a gap is created when the lead car is affected by a stop-and-go wave. The difference between these controls and the *PID* control, defined in Section 12.4.2, is that the autonomous vehicle never needs to catch up the lead vehicle and therefore, it does not create another stop-and-go wave for the controls defined in (12.19). More precisely, the term $\min(\max(\frac{\Delta x - 7\text{m}}{23\text{m}}, 0), 1)$ in (12.13) is not needed anymore. Moreover, in (12.18), we do not use that the length of the ring is equal to L.

The effect of the Lyapunov controllers can be seen in the velocity profiles in Figures 12.7 and 12.8 which shows that the traffic is smoother when the control is active. More precisely, in Figure 12.7, we notice that the AV tends to have very small oscillations when the control is active. On the other hand, the controlled vehicles tend to keep some velocity oscillations, however apparently less than the PID controller. Due to the presence of saturation, this control steers the systems toward a local equilibrium. In particular, for some initial data, the control dampens the stop-and-go waves without completely dissipating them. Figure 12.8 shows a behavior pretty similar to that of the PID controller, with even stronger oscillations for the AV.
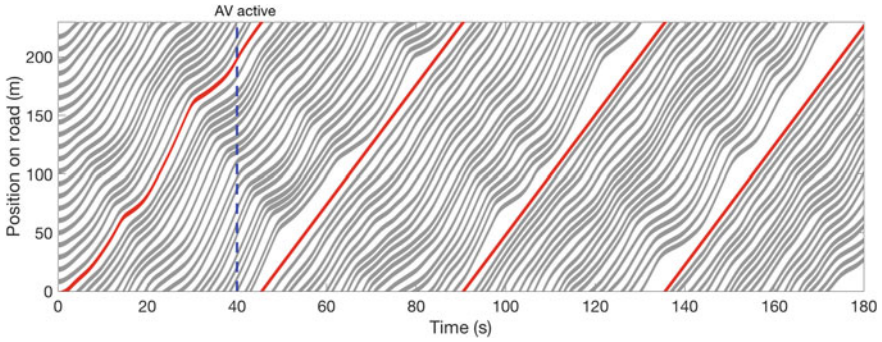
Fig. 12.5 Vehicles trajectories when using the control (12.19) with (12.16)
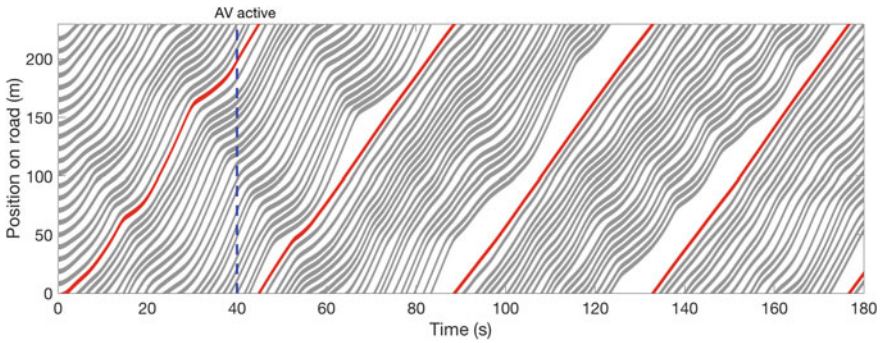

Fig. 12.6 Vehicles trajectories when using the control (12.19) with (12.17)
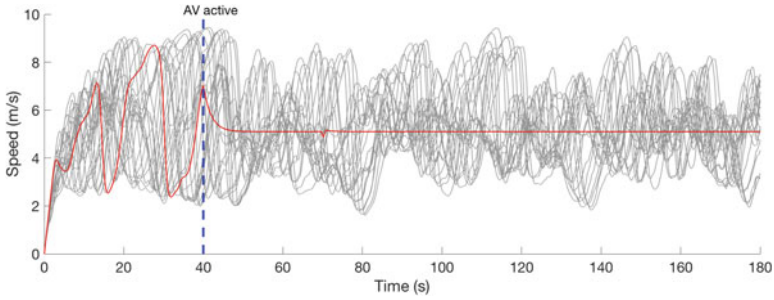

Fig. 12.7 Simulated velocity profiles of human-piloted vehicles (gray) and autonomous vehicle (red) when using the control (12.19) with (12.16)
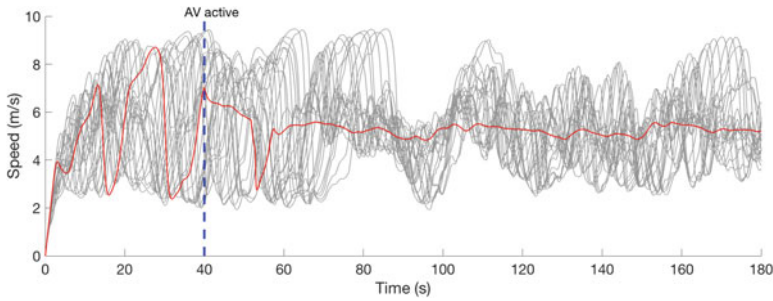
Fig. 12.8 Simulated velocity profiles of human-piloted vehicles (gray) and autonomous vehicle (red) when using the control (12.19) with (12.17)

12.4.4 Fuel Consumption

Fuel consumption is estimated for the simulation results using the VT-Micro fuel consumption model [45, 46]. The VT-Micro model uses polynomial regression on vehicle fuel consumption data collected at *Oak Ridge National Lab* to predict fuel consumption based on vehicle speed and acceleration. In simulation, per-vehicle fuel consumption averaged over an ensemble of 10 simulation runs when all traffic is under human control is 21.93 ℓ/100 km. When the *PID* controller (12.14) is implemented on the AV in simulation and the remaining 21 vehicles are under human control, the average per-vehicle fuel consumption is reduced by 16.3% to 18.36 ℓ/100 km. Similarly, when the controller in (12.19) with (12.16) is used on the AV, the average per-vehicle fuel consumption is reduced by 17.8% to 18.03 ℓ/100 km, while when the controller in (12.19) with (12.17) is used on the AV, the average per-vehicle fuel consumption is reduced by 17.3% to 18.13 ℓ/100 km. The fuel consumption means over the 10 simulation runs conducted and corresponding standard deviations are presented in Table 12.1. As described in Section 12.4.3, the Lyapunov controllers might not always dissipate the stop-and-go waves which result in a higher standard deviation for the mean fuel consumption in Table 12.1 for these controls.

Table 12.1 Simulation fuel consumption estimates over an ensemble of 10 simulation runs using the VT-Micro fuel consumption model

Control	Mean fuel consumption (ℓ/100 km)	Standard deviation (ℓ/100 km)
No control	21.93	0.94
PID controller (12.14)	18.36	0.35
(12.19) with (12.16)	18.03	0.46
(12.19) with (12.17)	18.13	0.65

12.5 Experimental Results

12.5.1 Experimental Design

A series of experiments were conducted to validate the development of stop-and-go traffic waves and to demonstrate the dissipation of these waves using the designed controller. We first describe the experiment setup and its benefits and limitations. Then we present the experimental results of a low penetration rate of autonomous vehicles actively dissipating stop-and-go waves.

The experiments are conducted using a similar setup to the seminal works of Sugiyama et al. [53] and Tadaki et al. [56], which was able to isolate human driving behavior without considering other factors. These experiments showed conclusively that human driving behavior alone is sufficient to trigger stop-and-go waves. In this experiment, we use the same setup because it has been shown to reliably produce the kinds of stop-and- go waves that the controllers in this paper are designed to dampen.

While the underlying experimental setup in the work presented is similar to the setup used by Sugiyama et al. [53], minor modifications were made to accommodate for the larger US vehicles. Just as in [53], the experiment is conducted with a total of 22 vehicles on the track. However, due to the substantially larger US vehicles, the track length was increased to 260 m around. While all vehicles have human drivers in them, one of the vehicles, the University of Arizona *Cognitive Autonomous Test Vehicle* (CAT Vehicle) is an autonomous-capable vehicle and can be switched from being human-piloted to autonomous during the experiment.

The experiment begins with all vehicles evenly spaced and at rest. When the drivers are given a signal, they begin to drive, and human-piloted traffic conditions are observed. After some time the autonomous driving capabilities of the CAT Vehicle are activated, and traffic with 21 human drivers and one autonomous vehicle is observed.

Vehicle trajectories of each vehicle were collected using a *VSN Mobile V360* panoramic video camera placed at the center of the track. The video footage recorded during the experiment was processed using image processing algorithms. More details on the image processing algorithms used can be found in the article by Wu et al. [59]. Additionally, vehicle performance data such as fuel consumption was recorded during the experiment using *OBDLink MX onboard diagnostics* (OBD-II) data loggers.

The ring-road experimental design is selected because it has been shown to produce traffic instabilities that are similar to those observed in real highway traffic [53]. Furthermore, using a closed-circuit experimental test track allowed us to reproduce "infinite" traffic (where each vehicle has a vehicle in front of it and a vehicle behind it at all times) with a finite number of vehicles. However, there are some limitations that arise due to this experimental design. Since the track is only a single lane of traffic, this experimental setup cannot be used to assess the robustness of the designed algorithm to overtaking and merging.

Fig. 12.9 Vehicles on test
track during the experiment

Fig. 12.10 The CAT
Vehicle, the autonomous
vehicle used during the
experiment

12.5.2   Experimental Results

An experiment was conducted to test the PID velocity controller in a field test. The experiment was conducted on a large flat parking lot in Tucson, AZ. The human-piloted vehicles used during this experiment were rented from the University of Arizona Motor Pool, while the AV that was used was the CAT Vehicle. An overview of the test track is seen in Figure 12.9 where all 22 vehicles are on the track following the experiment. The AV used to implement the traffic controller, the University of Arizona CAT Vehicle, is seen in Figure 12.10 being parked at the start position for the experiment.

The experiment is started with all vehicles at rest on the track and under human control. The velocity profile and vehicle trajectories for all vehicles in the experiment is presented in Figure 12.11 and Figure 12.12, respectively. Here, the CAT Vehicle's speed and trajectory are plotted in red, while the speed and trajectory for the remaining vehicles are plotted in gray. After 161 s the small oscillations grow, and a noticeable stop and go wave develops. In the presence of this wave, vehicles fluctuate in speed between 0 and 14 m/s as seen in Figure 12.11. This same stop-and-go wave is also seen in the vehicle trajectories in Figure 12.12. After 218 s the controller on the CAT Vehicle is activated and the traffic is under AV control. This control action is maintained until the experiment is ended after 413 s.

As seen in the vehicle trajectories in Figure 12.12, after roughly 320 s, one of the human-piloted vehicles introduces a small traffic wave. This wave propagates upstream until it encounters the CAT Vehicle, which is able to partially dampen, but
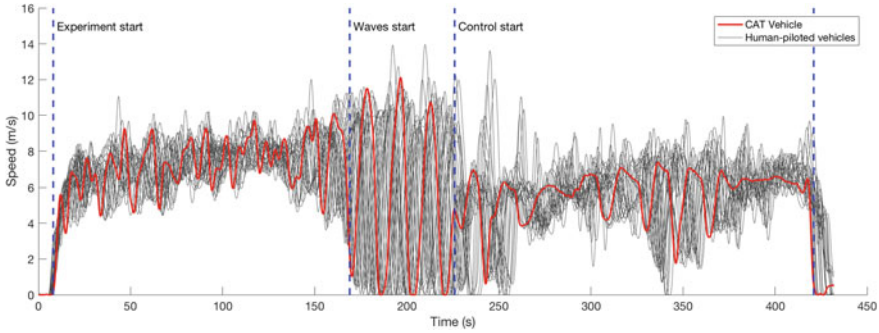
Fig. 12.11 Velocity profile of all vehicles in the experiment. The gray velocity profiles are the human-piloted vehicles and the red profile is the CAT Vehicle. The blue vertical lines mark key times during the experiment: the start, at what point waves are clearly visible, when control of the CAT Vehicle is activated, and the end of the experiment
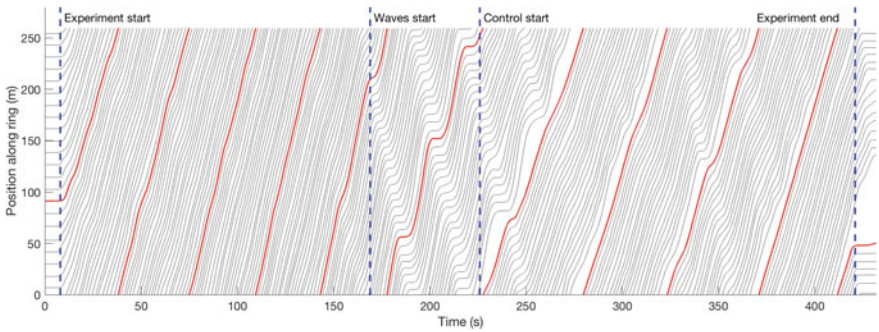

Fig. 12.12 Trajectories of all vehicles in the experiment. The gray trajectories are the human-piloted vehicles and the red trajectory is the CAT Vehicle. The blue vertical lines mark key times during the experiment: the start, at what point waves are clearly visible, when control of the CAT Vehicle is activated, and the end of the experiment

not fully remove the wave. On the second pass around the track, the CAT Vehicle is able to fully eliminate the wave. This demonstrates that a single AV may not be sufficient to fully eliminate stop-and-go waves, but a low penetration rate of AVs may be sufficient to eliminate, or at least substantially dampen stop-and-go waves.

The effect that the traffic controller implemented on the single AV has on the overall traffic flow (all 21 human-piloted vehicles in addition to the CAT Vehicle) is apparent in the velocity profile in Figure 12.11 and vehicle trajectories 12.12 where substantially smoother traffic is seen when the traffic is under the control of the AV. Specifically, the average velocity standard deviation between the time period where waves are present and the control period is reduced by 54.7% from 3.85 to 1.74 m/s while not substantially effecting throughput (1755 veh/h when waves are present and 1711 veh/h when the AV is actively dissipating the stop and go waves, or a reduction of 2.5%). Furthermore, since each vehicle in the flow was instrumented with an OBD-

II scanner, it was possible to measure the instantaneous fuel consumption of each vehicle during the experiment. The average fuel consumption over the entire fleet when waves are present was 26.3 ℓ/100 km, while the average fuel consumption of all vehicles when the AV is actively dissipating stop-and-go waves is reduced by 21.1% to 20.7ℓ/100 km. It is important to note that this reduction in fuel consumption is not only realized on the CAT Vehicle, but is the averaged quantity across all vehicles in the experiment.

This experimental result validates the theoretical and simulation-based results and demonstrates that even at a low penetration rate, AVs are capable of substantially improving traffic flow by reducing velocity standard deviation. This leads to smoother traffic and reduces fuel consumption of not only the AV, but all the vehicles in the traffic flow.

12.6 Conclusion

In this work, we establish controllability results for two well-known microscopic traffic flow models in the setting in which an autonomous vehicle is able to be controlled with the aim of dissipating stop-and-go traffic waves. Based on the controllability results, three control algorithms are developed and tested in simulation, indicating that fuel consumption reductions of up to 20% may be achieved when the flow is stabilized by the autonomous vehicle. In a first of its kind field test, we further establish for one of the proposed control algorithms that traffic wave dissipation is possible with real autonomous and human-piloted vehicles, and leads to a substantial reduction in fuel consumption compared to when waves are present.

While our results show the feasibility of control of traffic via AV, especially for fuel consumption reduction, there are some limitations to our study. For instance, multilane traffic was not considered and, more generally, we did not include additional challenges coming from the impact of combined complex phenomena of multilane dynamics, merges, ramps, and non-FIFO assumptions.

Looking forward, we note that the level of difficulty to prove controllability results of the combined optimal velocity follow the leader model for $n > 9$ suggests alternate modeling scales, e.g., in the micro-macro direction [6, 11, 28, 40], might be promising.

Appendix

The linearization of system (12.8) at the equilibrium point $(Y^*, 0) \in \mathbb{R}^{2n}$ is described by

$$\dot{Y} = AY + Bu, \tag{12.20}$$

where

$$A = \begin{pmatrix} \boxed{\begin{array}{cc} 0_{n-1} & I_{n-1} \\ D_{n-1} & F_{n-1} \end{array} \begin{array}{c} 0 \\ \vdots \\ \vdots \\ 0 \\ b \end{array}} \\ \boxed{0 \quad \cdots\cdots\cdots\cdots \quad \cdots \quad 0} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \\ 1 \end{pmatrix}. \tag{12.21}$$

For $n \in \mathbb{N}$ define the matrices $(D_{n-1}, F_{n-1}) \in \mathcal{M}_{2n-1}(\mathbb{R})^2$ by:

$$D_{n-1} = \begin{pmatrix} -\gamma & \gamma & & (0) \\ & -\gamma & \ddots & \\ & & \ddots & \gamma \\ (0) & & & -\gamma \end{pmatrix}, \quad F_{n-1} = \begin{pmatrix} -(\alpha+b) & \alpha & & (0) \\ & -(\alpha+b) & \ddots & \\ & & \ddots & \alpha \\ (0) & & & -(\alpha+b) \end{pmatrix}. \tag{12.22}$$

with $\gamma = bV'(y^* + d)$ and $\alpha = \frac{a}{(y^*+d)^2}$. Moreover, denote by $0_{n-1} \in \mathcal{M}_{n-1}(\mathbb{R})$ and $I_{n-1} \in \mathcal{M}_{n-1}(\mathbb{R})$ the zero matrix and the identity matrix respectively. Let $k \geqslant 1$. From (12.21), there exist $(A_{i,k})_{i \in \{1,\ldots,4\}} \in \mathcal{M}_{n-1}(\mathbb{R})$ and $(C_{i,k})_{i \in \{1,2\}} \in \mathcal{M}_{n-1,1}(\mathbb{R})$ such that,

$$A^k = \begin{bmatrix} A_{1,k} & A_{2,k} & C_{1,k} \\ A_{3,k} & A_{4,k} & C_{2,k} \\ 0_{1,n-1} & 0_{1,n-1} & 0 \end{bmatrix},$$

and for every $k \geqslant 1$ we have

$$\begin{cases} A_{4,k+1} = D_{n-1}A_{2,k} + F_{n-1}A_{4,k}, \\ A_{2,k+1} = A_{4,k}, \\ C_{2,k+1} = D_{n-1}C_{1,k} + F_{n-1}C_{2,k}, \\ C_{1,k+1} = C_{2,k}. \end{cases} \tag{12.23}$$

In particular, for every $k \geqslant 2$,

$$\begin{cases} A_{4,k+1} = D_{n-1}A_{4,k-1} + F_{n-1}A_{4,k}, \\ A_{2,k+1} = D_{n-1}A_{2,k-1} + F_{n-1}A_{2,k}, \\ C_{2,k+1} = D_{n-1}C_{2,k-1} + F_{n-1}C_{2,k}, \\ C_{2,k} = D_{n-1}C_{1,k-1} + F_{n-1}C_{1,k}. \end{cases} \tag{12.24}$$

Combining (12.23) with (12.24) and using that, for every $k \geqslant 1$, $A^k B(2n-1) = 0$, we conclude that, for every $k \geqslant 2$,

$$A^{k+1}B = \mathscr{D}A^{k-1}B + \mathscr{F}A^k B, \tag{12.25}$$

with

$$\mathscr{D} = \begin{bmatrix} D_{n-1} & 0_{n-1} & 0 \\ 0_{n-1} & D_{n-1} & 0 \\ 0_{1,n-1} & 0_{1,n-1} & 0 \end{bmatrix} \quad \text{and} \quad \mathscr{F} = \begin{bmatrix} F_{n-1} & 0_{n-1} & 0 \\ 0_{n-1} & F_{n-1} & 0 \\ 0_{1,n-1} & 0_{1,n-1} & 0 \end{bmatrix}. \tag{12.26}$$

Moreover, we have

$$A^2 B = X + \mathscr{F}_{2n-1}AB \quad \text{with} \quad AB = \begin{pmatrix} 0_{n-2,1} \\ 1 \\ 0_{n-3,1} \\ \alpha \\ -\alpha \\ 0 \end{pmatrix} \quad \text{and} \quad X = \begin{pmatrix} 0_{n-2,1} \\ b \\ 0_{n-3,1} \\ \gamma \\ -\gamma \\ 0 \end{pmatrix}. \tag{12.27}$$

Proof of Theorem 12.3 If $b = 0$ then $\gamma = 0$. From (12.25) and (12.27), for every $k \geqslant 2$,

$$\begin{cases} A^{k+1} = \mathscr{F}A^k B, \\ A^2 B = \mathscr{F}AB. \end{cases}$$

Thus, the Kalman controllability matrix (12.6) satisfies

$$\text{rank}\, (K(A, B)) = \text{rank}\, \big(B, AB, \mathscr{F}AB, \ldots, \mathscr{F}^{2n-4}AB \big).$$

By Cayley–Hamilton Theorem, there exists $(\alpha_0, \ldots, \alpha_{n-2}) \in \mathbb{R}^{n-1}$ such that $F_{n-1}^{n-1} = \sum_{i=0}^{n-2} \alpha_i F_{n-1}^i$. From (12.26), we conclude that

$$\text{rank}\, (K(A, B)) = \text{rank}\, \big(B, AB, \mathscr{F}AB, \ldots, \mathscr{F}^{n-2}AB \big).$$

Using the expression of AB given in (12.27) and the equality $\mathscr{F}^k = \begin{bmatrix} F_{n-1}^k & 0_{n-1} & 0 \\ 0_{n-1} & F_{n-1}^k & 0 \\ 0_{1,n-1} & 0_{1,n-1} & 0 \end{bmatrix}$,

by straightforward computations, we have

$$\text{rank}\, \big(B, AB, \mathscr{F}AB, \ldots, \mathscr{F}^{n-2}AB \big) = n,$$

whence the conclusion.

Proof of Theorem 12.4 Let's prove by induction that, for every $k \geqslant 1$, there exist $(\lambda_i)_{i=1,\ldots,2k}$ and $(\mu_i)_{i=1,\ldots,2k+1}$ such that

$$\begin{cases} A^{2k+1}B = \sum_{i=1}^{2k} \lambda_i A^i B + \mathscr{D}^k(AB) \\ A^{2k+2}B = \sum_{i=1}^{2k+1} \mu_i A^i B + \mathscr{D}^k(A^2 B) \end{cases} . \tag{P_k}$$

Since $a = 0$, we have $F_{n-1} = -b I_{n-1}$. Using (12.25), $A^3 B = \mathscr{D}AB - bA^2 B$ and $A^4 B = \mathscr{D}A^2 B - bA^3 B$. Thus, (P_k) holds for $k = 1$. Assuming that (P_k) holds for $k = p$. From (12.25), we have

$$\begin{cases} A^{2p+3} = \mathscr{D}A^{2p+1} - bA^{2p+2}B \\ A^{2p+4} = \mathscr{D}A^{2p+2} - bA^{2p+3}B \end{cases} \quad \text{and} \quad \mathscr{D}A^i B = A^{i+2}B + bA^{i+1}B, \; i \geqslant 1. \tag{12.28}$$

Using (12.28) and (P_k) for $k = p$, we conclude that (P_k) holds for $k = p + 1$. The equality (P_k) for $k = n - 2$ gives

$$\begin{aligned} \text{rank}(K(A, B)) &= \left(B, AB, \ldots, A^{2n-3}, \mathscr{D}^{n-2}A^2 B\right) \\ &= \left(B, AB, \ldots, A^{2n-4}, \mathscr{D}^{n-2}AB, \mathscr{D}^{n-2}A^2 B\right) \\ &= \left(B, AB, A^2 B, \mathscr{D}AB, \mathscr{D}A^2 B, \ldots, \mathscr{D}^{n-2}AB, \mathscr{D}^{n-2}A^2 B\right). \end{aligned}$$

Since AB and $A^2 B$ are linearly independent and $\mathscr{D}^k = \begin{bmatrix} D_{n-1}^k & 0_{n-1} & 0 \\ 0_{n-1} & D_{n-1}^k & 0 \\ 0_{1,n-1} & 0_{1,n-1} & 0 \end{bmatrix}$, by

straightforward computations, we have

$$\text{rank}\left(B, AB, A^2 B, \mathscr{D}AB, \mathscr{D}A^2 B, \ldots, \mathscr{D}^{n-2}AB, \mathscr{D}^{n-2}A^2 B\right) = 2n - 1.$$

Thus, the linearization of system (12.8) at the equilibrium point $(Y^*, 0)$ is controllable. Using [8, Theorem 3.8], Theorem 12.4 is proved.

Proof of Theorem 12.5 Using the symbolic mathematics software Maple [https://www.maplesoft.com/products/Maple/], we establish that for every $3 \leqslant n \leqslant 9$,

$$\text{Det}(K(A, B)) = \gamma^{\frac{n^2-3n+2}{2}} b^{\frac{n^2-n}{2}} \left(\alpha - \frac{\gamma}{b}\right)^{\frac{n^2-n}{2}}.$$

Thus, for every $\alpha \neq \frac{\gamma}{b}$, $\gamma \neq 0$, $b \neq 0$, the linearization of system (12.8) at the equilibrium point $(Y^*, 0)$ is controllable, whence the conclusion of Theorem 12.5 by using [8, Theorem 3.8].

References

1. Bando, M., Hasebe, K., Nakayama, A., Shibata, A., Sugiyama, Y.: Dynamical model of traffic congestion and numerical simulation. Phys. Rev. E **51**(2), 35–1042 (1995)
2. Bellomo, N., Dogbe, C.: On the modeling of traffic and crowds: a survey of models, speculations, and perspectives. SIAM Rev. **53**(3), 409–463 (2011)
3. Besselink, B., Johansson, K.H.: String stability and a delay-based spacing policy for vehicle platoons subject to disturbances. IEEE Trans. Automat. Control (2017)
4. Bressan, A., Piccoli, B.: Introduction to the mathematical theory of control. In: AIMS Series on Applied Mathematics. American Institute of Mathematical Sciences (AIMS), vol. 1. Springfield, MO (2007)
5. Buehler, M., Iagnemma, K., Singh, S.: The DARPA Urban Challenge: Autonomous Vehicles in City Traffic, vol. 56. Springer (2009)
6. Chalons, C., Delle Monache, M.L., Goatin, P.: A conservative scheme for non-classical solutions to a strongly coupled pde-ode problem. Interfaces Free Boundaries **19**(4), 553–571 (2017)
7. Chong, L., Abbas, M., Medina, A.: Simulation of driver behavior with agent-based back-propagation neural network. Transp. Res. Record J. Transp. Res. Board **2249**, 44–51 (2011)
8. Coron, J.-M.: Control and Nonlinearity. Number 136. American Mathematical Society (2007)
9. Darbha, S., Rajagopal, K.: Intelligent cruise control systems and traffic flow stability. Transp. Res. Part C Emerg. Technol. **7**(6), 329–352 (1999)
10. Davis, L.: Effect of adaptive cruise control systems on traffic flow. Phys. Rev. E **69**(6), 066110 (2004)
11. Delle Monache, M.L., Goatin, P.: Scalar conservation laws with moving constraints arising in traffic flow modeling: an existence result. J. Different. Equations **257**(11), 4015–4029 (2014)
12. Fenton, R.E., Mayhan, R.J.: Automated highway studies at the Ohio State University-an overview. IEEE Trans. Vehicular Technol. **40**(1), 100–113 (1991)
13. Garavello, M., Han, K., Piccoli, B.: Models for Vehicular Traffic on Networks. American Institute of Mathematical Sciences (2016)
14. Ge, G.I., Avedisov, S.S., He, C.R., Qin, W.B., Sadeghpour, M., Orosz, G.: Experimental validation of connected automated vehicle design among human-driven vehicles. Transp. Res. Part C Emerg. Technol. **91**, 335–352 (2018)
15. Gomes, G., Horowitz, R.: Optimal freeway ramp metering using the asymmetric cell transmission model. Transp. Res. Part C Emerg. Technol. **14**(4), 244–262 (2006)
16. Guériau, M., Billot, R., El Faouzi, N.-E., Monteil, J., Armetta, F., Hassas, S.: How to assess the benefits of connected vehicles? A simulation framework for the design of cooperative traffic management strategies. Transp. Res. Part C Emerg. Technol. **67**, 266–279 (2016)
17. Han, Y., Hegyi, A., Yuan, Y., Hoogendoorn, S., Papageorgiou, M., Roncoli, C.: Resolving freeway jam waves by discrete first-order model-based predictive control of variable speed limits. Transp. Res. Part C Emerg. Technol. **77**, 405–420 (2017)
18. Hegyi, A., De Schutter, B., Hellendoorn, H.: Model predictive control for optimal coordination of ramp metering and variable speed limits. Transp. Res. Part C Emerg. Technol. **13**(3), 185–209 (2005)
19. Hegyi, A., De Schutter, B., Hellendoorn, J.: Optimal coordination of variable speed limits to suppress shock waves. IEEE Trans. Intell. Transp. Syst. **6**(1), 102–112 (2005)
20. Hegyi, A., Hoogendoorn, S.P., Schreuder, M., Stoelhorst, H., Viti, F.: SPECIALIST: a dynamic speed limit control algorithm based on shock wave theory. In: Proceedings of the IEEE Conference on Intelligent Transportation Systems, pp. 827–832. IEEE (2008)
21. Helbing, D.: Traffic and related self-driven many-particle systems. Rev. Modern Phys. **73**, 1067–1141 (2001)
22. Hongfei, J., Zhicai, J., Anning, N.: Develop a car-following model using data collected by "five-wheel system". In: Proceedings of the IEEE Conference on Intelligent Transportation Systems, vol. 1, pp. 346–351. IEEE (2003)

23. Ioannou, P., Xu, Z., Eckert, S., Clemons, D., Sieja, T.: Intelligent cruise control: theory and experiment. In: Proceedings of the 32nd IEEE Conference on Decision and Control, pp. 1885–1890. IEEE (1993)
24. Jurdjevic, V., Quinn, J.P.: Controllability and stability. J. Different. Equations **28**(3), 381–389 (1978)
25. Khodayari, A., Ghaffari, A., Kazemi, R., Braunstingl, R.: A modified car-following model based on a neural network model of the human driver effects. IEEE Trans. Syst. Man Cybernet. Part A Syst. Humans **42**(6), 1440–1449 (2012)
26. Knorr, F., Baselt, D., Schreckenberg, M., Mauve, M.: Reducing traffic jams via VANETs. IEEE Trans. Vehicular Technol. **61**(8), 3490–3498 (2012)
27. Konishi, K., Kokame, H., Hirata, K.: Decentralized delayed-feedback control of an optimal velocity traffic model. Eur. Phys. J. B Condensed Matter Complex Syst. **15**(4), 715–722 (2000)
28. Lattanzio, C., Maurizi, A., Piccoli, B.: Moving bottlenecks in car traffic flow: a PDE-ODE coupled model. SIAM J. Math. Anal. **43**, 50–67 (2011)
29. Levine, W., Athans, M.: On the optimal error regulation of a string of moving vehicles. IEEE Trans. Automat. Control **11**(3), 355–361 (1966)
30. Lu, X.Y., Qiu, T.Z., Varaiya, P., Horowitz, R., Shladover, S.E.: Combining variable speed limits with ramp metering for freeway traffic control. In: Proceedings of the American Control Conference, pp. 2266–2271 (2010)
31. Lu, X.-Y., Shladover, S.E., Jawad, I., Jagannathan, R., Phillips, T.: Novel algorithm for variable speed limits and advisories for a freeway corridor with multiple bottlenecks. Transp. Res. Record J. Transp. Res. Board **2489**, 86–96 (2015)
32. Ma, J., Li, X., Shladover, S.E., Rakha, H.A., Lu, X.-Y., Jagannathan, R., Dailey, D.J.: Freeway speed harmonization. IEEE Trans. Intell. Vehicles **1**(1), 78–89 (2016)
33. Milanés, V., Shladover, S.E., Spring, J., Nowakowski, C., Kawazoe, H., Nakamura, M.: Cooperative adaptive cruise control in real traffic situations. IEEE Trans. Intell. Transp. Syst. **15**(1), 296–305 (2014)
34. Nakayama, A., Sugiyama, Y., Hasebe, K.: Effect of looking at the car that follows in an optimal velocity model of traffic flow. Phys. Rev. E **65**(1), 016112 (2001)
35. Nissan, A., Koutsopoulos, H.N.: Evaluation of the impact of advisory variable speed limits on motorway capacity and level of service. Procedia Soc. Behavior. Sci. **16**, 100–109 (2011)
36. Orosz, G., Wilson, R.E., Stépán, G.: Traffic Jams: Dynamics and Control (2010)
37. Papageorgiou, M., Hadj-Salem, H., Blosseville, J.-M.: ALINEA: a local feedback control law for on-ramp metering. Transp. Res. Record J. Transp. Res. Board **1320**, 58–67 (1991)
38. Papageorgiou, M., Kotsialos, A.: Freeway ramp metering: an overview. IEEE Trans. Intell. Transp. Syst. **3**(4), 271–281 (2002)
39. Papamichail, I., Kampitaki, K., Papageorgiou, M., Messmer, A.: Integrated ramp metering and variable speed limit control of motorway traffic flow. IFAC Proceedings Volumes **41**(2), 14084–14089 (2008)
40. Piacentini, G., Goatin, P., Ferrara, A.: Traffic control via moving bottleneck of coordinated vehicles. In: 15th IFAC Symposium on Control in Transportation Systems (2018)
41. Piccoli, B., Tosin, A.: Vehicular Traffic: A Review of Continuum Mathematical Models, pp. 1748–1770. Springer, New York (2011)
42. Pipes, L.A.: An operational analysis of traffic dynamics. J. Appl. Phys. **24**(3), 274–281 (1953)
43. Popov, A., Hegyi, A., Babuška, R., Werner, H.: Distributed controller design approach to dynamic speed limit control against shockwaves on freeways. Transp. Res. Record J. Transp. Res. Board **2086**, 93–99 (2008)
44. Rajamani, R., Choi, S.B., Hedrick, J.K., Law, B., Hedrick, B., Prohaska, R., Kretz, P.: Design and experimental implementation of control for a platoon of automated vehicles. AMSE J. Dynam. Syst. Meas. Control **122**(3), 470–476 (1998)
45. Rakha, H., Ahn, K., Trani, A.: Comparison of mobile5a, mobile6, vt-micro, and cmem models for estimating hot-stabilized light-duty gasoline vehicle emissions. Canadian J. Civil Eng. **30**(6), 1010–1021 (2003)

46. Rakha, H., Ahn, K., Trani, A.: Development of VT-Micro model for estimating hot stabilized light duty vehicle and truck emissions. Transp. Res. Part D Transp. Environ. **9**(1), 49–74 (2004)
47. Reuschel, A.: Vehicle movements in a platoon. Oesterreichisches Ingenieur-Archir **4**, 193–215 (1950)
48. Reuschel, A.: Vehicle movements in a platoon with uniform acceleration or deceleration of the lead vehicle. Zeitschrift des Oesterreichischen Ingenieur-und Architekten-Vereines **95**, 50–62 (1950)
49. SAE. Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems. Technical Report (2014)
50. Shladover, S.E.: Review of the state of development of advanced vehicle control systems (AVCS). Vehicle Syst. Dynam. **24**(6–7), 551–595 (1995)
51. Smulders, S.: Control of freeway traffic flow by variable speed signs. Transp. Res. Part B Methodologic. **24**(2), 111–132 (1990)
52. Sugiyama, Y., Fukui, M., Kikuchi, M., Hasebe, K., Nakayama, A., Nishinari, K., Tadaki, S., Yukawa, S.: Traffic jams without bottlenecks—experimental evidence for the physical mechanism of the formation of a jam. New J. Phys. **10**(3), 033001 (2008)
53. Sugiyama, Y., Fukui, M., Kikuchi, M., Hasebe, K., Nakayama, A., Nishinari, K., Tadaki, S.-I., Yukawa, S.: Traffic jams without bottlenecks-experimental evidence for the physical mechanism of the formation of a jam. New J. Phys. **10**(3), 033001 (2008)
54. Swaroop, D., Hedrick, J.: String stability of interconnected systems. IEEE Trans. Automat. Control **41**(3), 349–357 (1996)
55. Tadaki, S., Kikuchi, M., Fukui, M., Nakayama, A., Nishinari, K., Shibata, A., Sugiyama, Y., Yosida, T., Yukawa, S.: Phase transition in traffic jam experiment on a circuit. New J. Phys. **15**, 103034 (2013)
56. Tadaki, S.-I., Kikuchi, M., Fukui, M., Nakayama, A., Nishinari, K., Shibata, A., Sugiyama, Y., Yosida, T., Yukawa, S.: Phase transition in traffic jam experiment on a circuit. New J. Phys. **15**(10), 103034 (2013)
57. Talebpour, A., Mahmassani, H.S.: Influence of connected and autonomous vehicles on traffic flow stability and throughput. Transp. Res. Part C Emerg. Technol. **71**, 143–163 (2016)
58. Wang, M., Daamen, W., Hoogendoorn, S.P., van Arem, B.: Cooperative car-following control: distributed algorithm and impact on moving jam features. IEEE Trans. Intell. Transp. Syst. **17**(5), 1459–1471 (2016)
59. Wu, F., Stern, R., Cui, S., Monache, M.L.D., Bhadanid, R., Bunting, M., Churchill, M., Hamilton, N., Haulcy, R., Piccoli, B., Seibold, B.: Tracking vehicle trajectories and fuel rates in oscillatory traffic. Transp. Res. Part C Emerg. Technol. Under Review (2017)
60. Zhou, M., Qu, X., Li, X.: A recurrent neural network based microscopic car following model to predict traffic oscillation. Transp. Res. Part C Emerg. Technol. **84**, 245–264 (2017)

Chapter 13
Disturbance Rejection Run-to-Run Controller for Semiconductor Manufacturing

Marzieh Khakifirooz, Mahdi Fathi and Panos M. Pardalos

Abstract This chapter introduces a framework of disturbance rejection controller for discrete-time Run-to-Run (R2R) control system in semiconductor manufacturing environments. While we discussed the source of uncertainty and disturbance in wafer fabrication process, the photolithography process as one of the cutting-edge steps in wafer fabrication is selected for illustrating the power of disturbance rejection algorithm for compensating the misalignment. Along with this case study, some classification of disturbance rejection control algorithm with the structure of control plant is discussed.

13.1 Introduction

As society explores the Fourth Industrial revolution characterized by access to and leveraging of knowledge in the manufacturing enterprise, a meticulous and intelligent process control is needed to achieve higher throughput and customer satisfaction [31]. Controlling a complex system is challenging because the process components and variables operate autonomously and interoperate with other manufacturing segments. The immense in manufacturing complexity causes the several sources of measurable and unmeasurable uncertainties such as disturbance. This chapter aims to tackle the disturbances in feedback control operation in semiconductor production process by engaging the disturbance rejection models into the system.

M. Khakifirooz (✉)
Tecnológico de Monterrey, Monterrey, Mexico
e-mail: mkhakifirooz@tec.mx

M. Fathi
Mississippi State University, Starkville, MS, USA
e-mail: fathi@ise.msstate.edu

P. M. Pardalos
University of Florida, Gainesville, FL, USA
e-mail: pardalos@ise.ufl.edu

The objective of this work is to introduce readers the traditional and novel disturbance rejection systems for controlling the semiconductor manufacturing process. The paper then emphasizes on design and structure of control system with R2R architecture. It covers the technological foundations of feedback control system and addresses current threats faced by process engineers for handling the uncertainty of controlling the semiconductor production process along with existing state-of-the-art solutions for building up the disturbance-free optimization models. The topic will discuss from the perspectives of both practical implementations in the industry and cutting-edge academic research to provide a holistic mindset for process engineers and quality managers in industry, in addition to researchers and educators in the design and manufacturing communities.

The scope of this study is to build essential knowledge around control process in the semiconductor industry, the R2R control system and the disturbance rejection model, and other essentials. However, we are focusing almost exclusively on issues relevant to designing, constructing, and adapting the various disturbance rejection (free) control system for semiconductor manufacturing based on R2R control structure and optimization algorithm.

The remainder of this study is organized as follows, Section 13.2, introduces the structure of R2R feedback control system in the semiconductor industry. Section 13.3 discusses the source of uncertainties in wafer fabrication process. Section 13.4 introduces the design of disturbance rejection controllers including the structure of closed-loop system and algorithm of adaptive and robust control systems. Section 13.5, illustrates the control process of Photolithography process as one of the cutting-edge steps in wafer manufacturing and the case study of overlay error. The main challenges in Photolithography control process will discuss, and the result of reinforcement learning disturbance rejection model with traditional Exponentially Weighted Moving Average (EWMA) model will compare for further interpretation. Section 13.6, concludes the paper.

13.2   Run-to-Run Control System

The R2R control is one of the general controlling techniques in semiconductor manufacturing [38]. The primary objective of R2R control is variability reduction of the process through the shrinking the process output error. R2R control has been extensively adapted to analyze a variety of challenges in the process control of complex semiconductor manufacturing.

R2R control consists of two major steps:

1. building a linear regression model based on offline experiments between the input variable(s) u_t and the output or response variable(s) y_t.
2. estimating the process variable on online experiments, while the offline model based on the observed process and data is continuously updated and determine the control action (tuning the online model).

In semiconductor manufacturing, the number of runs for a specific product, recipe, chamber, and tool is small and collecting enough observations to fit and eventually use a model for control purposes is impossible [28]. Therefore, the control process is required the online estimation and tuning to predict the parameter setting of controller. In this case, at each machine's utilization time (in brief called run), the updated model is used to compute the control action.

Consider a single-input-single-output (SISO) control system, the basic assumption in the first step of R2R control is that the process exhibits static. This means that the output variable y_t at run t depends only on the input variable u_{t-1} at the end of run $t-1$ (when the inputs variable u_{t-1} or process output y_{t-1} at run $t-1$ has an effect on y_t the process exhibits dynamics).[1] The next assumption in the first step of R2R is that the process is modeled by fitting (optimizing) to the simple first-order linear process of the form

$$y_t = \alpha + \beta u_{t-1} + \varepsilon_t \tag{13.1}$$

where α and β are process intercept and gain (slope) parameters, ε_t is a white noise error.

After optimization within the first step, the second step tries to maintain the process variables as close as possible to the optimum (target) value. In R2R control system usually, EWMA filter is used for predicting error and feedback signal. The optimal control action for reaching the desired target T value for the process (13.1) is

$$\hat{u}_1 = \frac{T - \alpha_0}{\beta_0}. \tag{13.2}$$

where α_0 and β_0 are the initial values of α and β, respectively. Due to dynamic behavior of semiconductor industry, the basic assumption in R2R control is to having a time-varying (dynamic) intercept, α. Therefore, resulting from Eq. (13.2), the control action computes from predicted response value $\hat{y}_t = a_{t-1} + b\hat{u}_{t-1}$, where $b = \hat{\beta}$ as the gain of slope parameter estimates offline and $a_t = \hat{\alpha}_{t+1|t}$ computes recursively based on the EWMA equation

$$a_t = \lambda(y_t - b\hat{u}_{t-1}) + (1 - \lambda)a_{t-1}. \tag{13.3}$$

where λ is EWMA weighting. Figure 13.1 illustrates a block diagram of the general structure of an R2R controller. R2R controller consists of two major steps. First, at each run, a linear regression model is built to estimate the output measurement (inner loop in Fig. 13.1). The estimated model by inner loop is continuously updated and tuned based on output measurement data by the outer loop performing as a supervisor of the inner loop. In fact the outer loop takes post-process measurements and gives a control action for each run.

---

[1]Static model: $y_t = F(u_{t-1})$; Dynamic model: $y_t = F(y_{t-1}, u_{t-1})$.
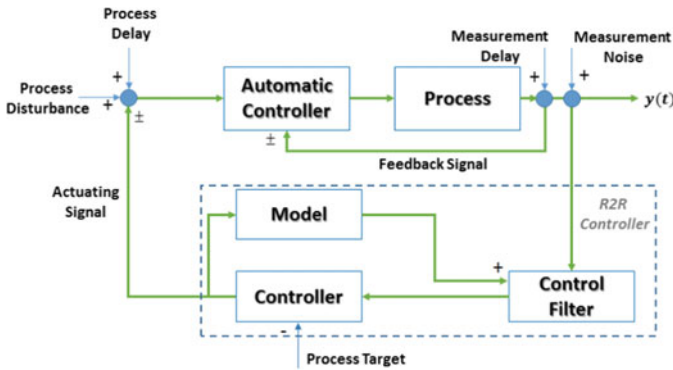
Fig. 13.1 Supervisory Run-to-Run controller

The EWMA controller is the most preferred design in R2R control. However, EWMA controller has several limitations. Some of the underlying limitations of EWMA include

- dependency on maximum likelihood estimation (MLE) of optimal decay factor,
- dependency on limited control action by fixed filtering parameters,
- dependency on multiple filtering steps,
- inefficiency to deal with the large-scale disturbance of the real-world system.

On the other hand, the traditional EWMA controller cannot satisfy the demand for high-mixed manufacturing process. Therefore, EWMA controller is not the best choice for applying in real-world case studies [29].

13.3   The Source of Disturbance in Semiconductor Industry

In wafer fabrication process, the growth or expansion of uncertainty about the health of processes and products often leads to major scrap events [28]. Thousands of products can be scrapped and generate a major production disturbance. Therefore, quality controls are required to take place at every stage of the production line, for protecting manufacturing systems from tool drift.

Industrial process models need to be identified and estimated from operating data and therefore encompass some level of uncertainty. Process variation can be caused by unmeasured disturbances (apparent in statistical models as random errors) or result from the uncertainty in model parameters, which are estimated from the data. In addition to these sources of variation, some process disturbances can be measured. However, the measured disturbances, such as process delay, are uncontrollable during actual production. Ambient temperatures and raw material variations can count as two typical examples of noise variables, which are encountered in manufacturing. Roughly speaking, uncertainty is divided into two categories:

- uncertainty in model parameters,
- uncertainty in noise variables.

Therefore, when considering a state-space system, the equation for predicting the system output, concerning the system noise and disturbance, is described as follows:

$$\begin{aligned} x_{t+1} &= C_1 x_t + C_2 u_t + d_t \\ y_{t+1} &= C_3 x_{t+1} + \varepsilon_t \end{aligned} \tag{13.4}$$

where d_t is process disturbance, x_t is states vector in state-space model for run t and C_1, C_2, and C_3 are coefficient matrices in state-space model, and y_t, u_{t-1} and ε_t are the same as (13.1).

Several studies have been implemented to reduce the effect of uncertainty on the performance of the control system. However, providing a robust solution to deal with both online variabilities of uncontrollable noise, and uncertainty in model parameters, is still an interesting and challenging research topic. Figure 13.1, merged the dynamic control system by uncertainties such as measurement noise, disturbance, process delay, and measurement delay.

13.3.1 Process Disturbance

Knowing the basics of system disturbances will assist control systems in the identification and controlling of such disturbances, which are representative of unusable parts of the actual value produced from any closed-loop system. Therefore, the efficient manner to avoid, and eliminate disturbances in the systems, is to use system feedback. The feedback loop assists to enable the control system to monitor the disturbances and processing system, to reduce, minimize, or eliminate disturbances, and achieve a state of stability in the system.

An integrated moving average ($IMA(1, 1)$) disturbance model, is widely employed in the control of discrete-part manufacturing processes [33]. Consider a process operates in closed-loop under a feedback control system. Feedback control can be employed to minimize the variability of outputs caused by invisible disturbances. In contrast, feedforward control can be used to decrease the influence of uncontrollable variables that have been measured, and which influence process variability. Sometimes, the prior knowledge as a form of Bayesian information can proceed to demonstrate the closed-loop identification.

13.3.2   Process Noise

In manufacturing processes, there are frequently, observable disturbances that can be measured during operations. Observable, but uncontrollable variables, are referred to as noise variables in process optimization literature.

Noise is an outcome of using sensor technologies or measuring process variables. Concerning electrical sensors and signals, measurement noise is often produced due to interaction with other electrical sources. Also, some physical blocking can affect sensors, resulting in incorrect signals being detected by the controller. In a typical process control situation, a proportional–integral–derivative (PID) controller can make corrective actions by reducing the proportion of the error between the process variable and the setpoint, combined with the integral and derivative, of that difference. The derivative action is most often affected by noise and disturbances.

In statistical process optimization literature, the statistical inference solutions are widely applied offline and are therefore not able to process adjustment methods, so that different controllable variable settings can be recommended, which are dependent on online noise variables and measured during production.

13.3.3   Stochastic Process Delay

The implementation of advanced process control (APC) in semiconductor manufacturing, blended with an inherent problem known as metrology delay which adversely affects R2R control performance.

Due to the need for the provision of rapid feedback to the process control, the lack of real-time metrology data causes extensive limitations in the R2R control. Most semiconductor manufacturing processes suffer from issues caused by metrology delays due to the time needed for measurements, metrology capacity, and the waiting time in the wafer queue between the processing tool and the metrology station [18]. The stability and performance of the process will be affected by the metrology delay. Moreover, since quality measurements perform online, the delay would not be fixed but flows stochastically. Several other phenomena make sever disturbance including the delay of proceeding because of bottleneck tools/processes in the system.

Thus far, numerous researches have been done in the semiconductor industry, to study on methods to reduce the effects of infrequent measurements and extensive metrology delays [16, 17, 51, 52, 55]. Virtual Metrology (VM) is deemed as the most popular technique and is a potential solution for overcoming these difficulties [34, 53].
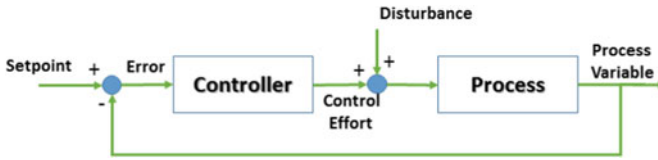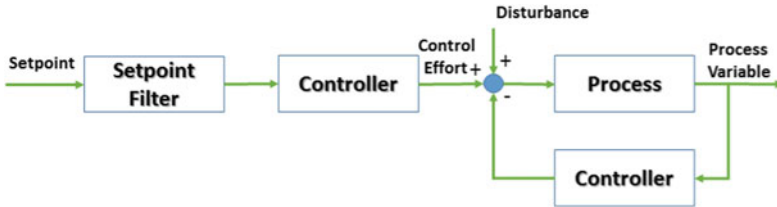
Fig. 13.2   Simple closed-loop operations diagram

Fig. 13.3   Closed-loop operations diagram with setpoint tracker

13.4   Design of Disturbance Rejection Control System

13.4.1   Structure of Disturbance Rejection Control Systems

The closed-loop control design is the best-suited design for disturbance rejection controllers.

Consider a simple feedback controller in closed-loop mode. When a disturbance added to the system, the process variable will begin to change according to the magnitude of the load and physical characteristics of the process. However, a simple feedback controller cannot determine how the process reacts to a disturbance, because the process response to disturbance is faster than its response to setpoint change. (See the simple closed loop operations diagram in Fig. 13.2).

Regards to the dependency of the closed-loop controller to the feedback signal, this question may come to mind "what happens to the stability of system when the feedback is inadequate?" This problem can be solved when the controller can be equipped with setpoint-filtering. The setpoint-filtering allows that mathematical inertia to minimize the distance between the setpoint and output variable. (See the Closed-Loop operations diagram in Fig. 13.3).

13.4.2   Algorithms for Disturbance Rejection Control System

In control system, the algorithm mainly classifies into robust and adaptive controllers. Robust systems are expected to work well with plants which change their characteristics along time, or in noisy environments and have fixed control parameters. However,

Table 13.1 Categories of feedback adaptive and robust control

Adaptive control	Robust control
Optimal dual controllers [24]	Sliding mode control [49]
Suboptimal dual controllers [1]	Variable structure control [4, 45]
Adaptive pole placement [9]	Linear–quadratic–Gaussian control [47]
Extremum-seeking controllers [46]	Active Disturbance Rejection Control [26]
Iterative learning control [3]	Passivity based control [39]
Gain scheduling [42]	Lyapunov based control [22]
Model reference adaptive controllers [10]	Quantitative feedback theory [41]
Model identification adaptive controllers [43]	Tracking differentiator [20]
Multiple models [25, 54]	Extended state observer [21]

it is implied that those changes are somewhat bounded and the closed-loop system which encompasses the fixed robust controller presents stability, controllability, and observability.

But in scenarios where the changes in the plant are extensive, often it is not possible to design a robust controller with a fixed parameter. In this case, one uses an adaptive controller whose parameters change with time and tracks the changes in the plant, with the goal of designing a system which performs by the design constraints.

In other words, an adaptive controller has to adapt to unknown uncertainties while a robust controller has to work within a compact set known a priori of uncertainties. The goal of an adaptive controller is to estimate unknown parameters first, usually online, and then derive the control law, while the purpose of a robust controller is to formulate a control law, usually based on worst case scenario, so that the controller works for the whole range of norm-bounded disturbance, without changing the control law.

Therefore, an adaptive controller adapts to the changes in its environment and modifies the control law based on the same information, while a robust controller provides a control law that is guaranteed to work throughout the norm-bounded disturbance range, without ever changing the control law.

Adaptive and robust control comes in many variants some of them along with some references are summarized in Table 13.1.

Adaptive or robust control methods are less successful when facing dynamicity (i.e., unbounded noise, change in environmental setting) or in real time obtaining missing information (i.e., delay). In these cases, the use of artificial intelligence (AI) tools can help to expand the capacity of complex control systems by covering tasks which quantitative models enable or less efficient to solve them. A variety of artificial intelligence tools can be used individually as the control system or as an auxiliary aid for quantitative models such as: neural network control [14], Bayesian control [23], fuzzy control [56], neuro-fuzzy control [27], expert systems [15], genetic control [35], and cognitive/conscious control [12]. Figure 13.4 illustrates the implementation of levels of different control models in a complex system.
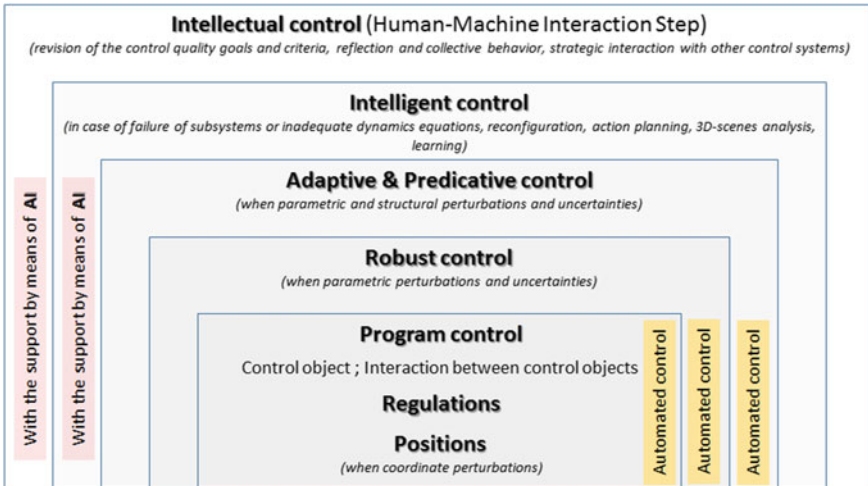
Fig. 13.4 Heterogeneous control of complex systems [50]

13.5   Case Study: Semiconductor Manufacturing and Challenges in Controlling the Patterning Process

Semiconductor device fabrication is encompassed in several processing steps. These steps are characterized by four primary sections, including patterning, etching/removal, deposition, and modification, as summarized in Table 13.2. Among these processes, the lithography process is the primary step in wafer fabrication. Following this section, some key demand features of lithography process are introduced for the analytical tools utilized in designing an adequate control system.

13.5.1   Photolithography Process

Lithography is one of the frequently used processes in fabricating chips, and typically between 30 and 35%, of the overall processing costs, and between 40 and 50%, of the completion time, is accounted for this process [37]. Additionally, the development plan for the future of lithography process required shrinkage in die size, and therefore lithography will have a technical limitation tendency when associated with the feature size reduction phenomena. Thereupon, lithography requires a high resolution, high sensitivity, precise alignment, and low defect density to achieve visions of wafer manufacturing. Therefore, setting up an accurate control system with high impact on disturbance rejection is an essential appliance for the lithography process.

Currently, the step-and-scan (shortened scanner) method is one of the most commercially used systems, in the lithography process. The purpose of the scanner is

Table 13.2 Semiconductor device fabrication steps

Process	Methods
Deposition: Grows, coats, or transfers a material onto the wafer	Physical vapor deposition
	Chemical vapor deposition
	Electrochemical deposition
	Molecular beam deposition
	Atomic layer deposition
Removal: Removes material from the wafer either in bulk mode or selectively	Wet etching
	Dry etching
	Chemical–mechanical planarization
Patterning: Shapes or alters the shape of the deposited materials	Lithography
Modification: Doped transistor sources and drains	Ion implantation
	Rapid thermal anneals
	Ultraviolet light processing

Fig. 13.5 Wafer fabrication in photolithography process
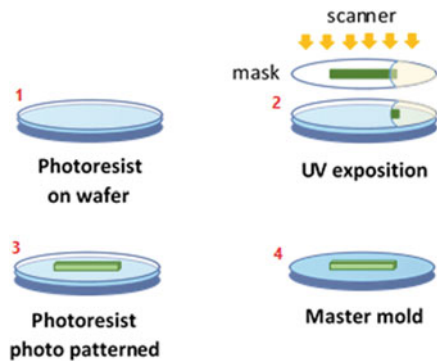
to superimpose a masking pattern on top of the existing wafer pattern. Figure 13.5 illustrates the initial steps of wafer fabrication in the photolithography process when using the scanner. The gap between the actual position of the mask over the actual position of the wafer substrate is known as the overlay error [5, 6]. As is evident from research, the overlay error has proven to be the most challenging issue in the photolithography process [2, 6, 40, 44].

13.5.1.1 Overlay Error

As discussed earlier, during semiconductor fabrication process, each wafer goes several times under the photolithography process, and at each time a layer of photo-

Fig. 13.6 The overlay error measurement [8]
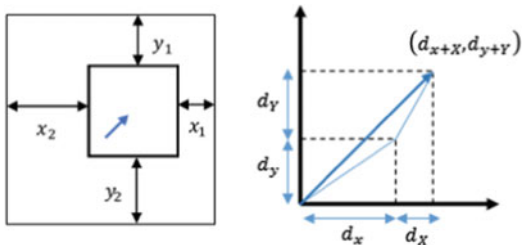
resist material exposure on the surface of wafer. The misalignment between the current and previous exposure layers, through the box-in-box design is called overlay error. When the inside box is accurately patterned in the center of the outside box, no overlay errors are apparent (Fig. 13.6).

The response variables of the overlay error, are indicated as follows:

$$d(x + X) = \frac{x_1 - x_2}{2}$$
$$d(y + Y) = \frac{y_1 - y_2}{2} \tag{13.5}$$

where (x, y) is intrafield coordinate system, regarding the center of the field[2] and (X, Y) is interfield coordinate system, regarding the center of the wafer. In Fig. 13.6, dx and dy denote to interfield overlay error in x and y direction, respectively, and dX and dY denote to interfield overlay error in X and Y direction, respectively.

The presence of overlay errors can be attributed to intrafield and interfield errors [7]. The interfield overlay errors are the result of the mismatch between the patterning mask and wafer. The intrafield errors are due to fitment problems between the lens of the scanner (light source), and the patterning mask. The interfield errors are measured at the center of the wafer, and the intrafield errors at the center of the exposure. The variables leading to intrafield, and interfield overlay errors, are presented in Fig. 13.7 and Fig. 13.8, respectively.

Various feedback controllers are designed based on R2R control for compensating the misalignment during the photolithography process. The most commonly applied and theoretical method is formed on EWMA estimation, other learning-based models are Kalman filters [11], artificial neural networks [32], machine learning [30], and PID controller [8]. Following this study, we introduce a new approach for compensating overlay error which has an advantage for faster disturbance rejection in comparison to the EWMA, PID, or Kalman filters based on reinforcement learning optimization.

---

[2]The surface of a wafer can be partitioned into smaller part for increase the accuracy of measurement the overlay error, each partition is called a field.
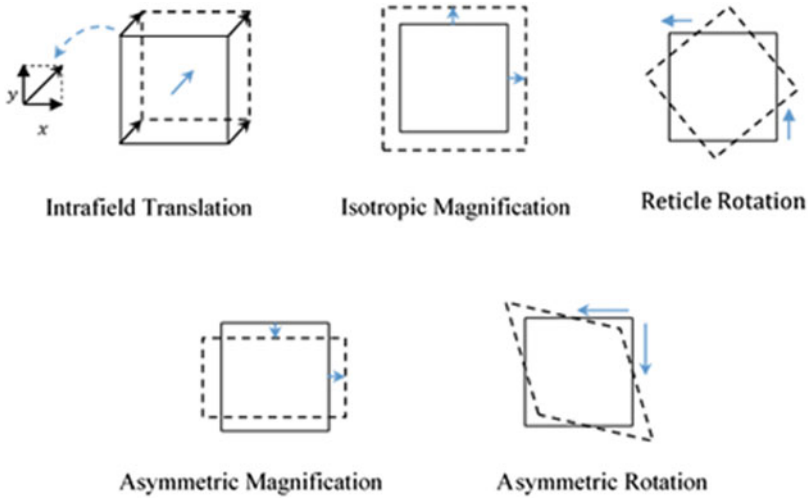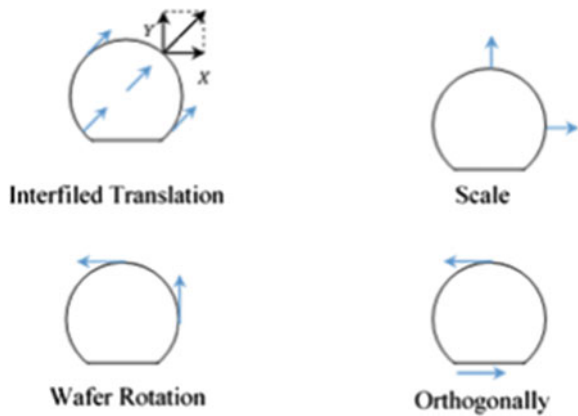
Fig. 13.7 Variables leading to intrafield overlay error [7]

Fig. 13.8 Variables leading to interfield overlay error [7]

13.5.2 Reinforcement Learning Optimization

A partially observable Markov decision process (POMDP) [36] is a generalization of Markov decision process (MDP) [48] which only part of the information is available about the current state, and this led to the uncertainties (i.e., delay, noise).

Consider a class of algorithms for finding good approximations to a class of learning problems in which agents interact in a dynamic, noisy and stochastic environment; this interaction is conventionally modeled as an POMDP which consists of

- S_t: a finite set of states.
- A_t: a finite set of actions.
- $R(S_t, A_t)$: a reward function.

- $P(S_{t+1}|S_t, A_t)$: a state transition probability function.
- O_t: a set of observations.
- $P(O_{t+1}|S_{t+1}, A_t)$: an observation probability.
- $r_t \in [0, 1]$: a discount factor.
- B_t: a distribution over state S_t called "Belief State".

POMDP can identify as an optimal or near-optimal behavior for an uncertain system. The MDP problem seeks to find a mapping from states to actions; however, the challenge in POMDP problem is to find a mapping from probability distributions over states to actions. For achieving this purpose, the key step is to calculate the value function of a given policy (π) which is the mapping function from the state to the action, to maximize the expected sum of the discounted factor. Regards to the definition of POMDP components described in above the optimization procedure of POMDP is following below steps:

1. set up the unobserved state S_t of the system at each time step t
2. select an action A_t,
3. maintain the distribution over S_t as B_t,
4. receive the reward function $R(S_t, A_t)$,
5. transitions to unobserved state S_{t+1} with probability $P(S_{t+1}|S_t, A_t)$,
6. receive an observation O_{t+1} with probability $P(O_{t+1}|S_{t+1}, A_t)$,
7. estimate a distribution on state S_{t+1} as $B_{t+1}(S_{t+1}) = P(S_{t+1}|O_{t+1}, A_t, B_t(S_t))$,
8. update the reward function by $R(S_{t+1}, A_{t+1}) = B_{t+1}(S_{t+1}) \times R(S_t, A_t)$
9. optimize the return function by policy $\pi(S_{t+1}) = \max_{A_{t+1}} \sum r_{t+1} R(S_{t+1}, A_{t+1})$ and select the best action A_{t+1},
10. update and repeat the process.

The Bellman's optimality [19] equation says that under principal of stochastic approximation, the average bias $Q(S_t, A_t)$ (Q-learning) from t times simulation-based solution is

$$Q(S_t, A_t) = \min_{A_t} \left[R(S_t, A_t) - \frac{1}{t} E \left\{ \sum_t R(S_t, A_t) \right\} + \sum_t P(S_t|S_{t-1}, A_{t-1}) \right.$$
$$\left. \times E \left\{ \sum_t R(S_t, A_t) - \frac{1}{t} E \left\{ \sum_t R(S_t, A_t) \right\} \right\} \right] \qquad (13.6)$$

The POMDP relies on defining the set of states and their expected values, the action transition matrix and reward structure. The description, role, and action of

each component in details for the control system of overlay model are summarized as follows[3]:

The state-space: The output of the controller to the plant ($\mathbf{x}_t$), given the action ($\mathbf{u}_t$), including the disturbance and process delay. Considering N overlay factors, therefore, there are $1 \times N$ different states in the system. Note that in practice the actual value of $\mathbf{x}_t$ is predictable, and not observable.

The observation space: The actual output of the plant ($\mathbf{y}_t$) for N overlay factors including the metrology delay and measurement noise.

The action transition matrix: The matrix of probability which each sate ($\mathbf{x}_t$) can appear in the sequence of the lithography process. Respecting the definition of variables in (13.4) the elements of transition matrix can be derived by

$$P(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_{t-1}). \tag{13.7}$$

The transition matrix can be computed based on the historical data and update after each run.

Belief updating: The probability distribution over $\mathbf{x}_t$ given the state of previous belief and observation and action at the current run.

$$B(\mathbf{x}_t) = \frac{P(\mathbf{y}_{t-1}|\mathbf{x}_{t-1},\mathbf{u}_{t-1})\sum_t P(\mathbf{x}_t|\mathbf{x}_{t-1},\mathbf{u}_{t-1})B(\mathbf{x}_{t-1})}{\sum_t P(\mathbf{y}_{t-1}|\mathbf{x}_{t-1},\mathbf{u}_{t-1})P(\mathbf{x}_t|\mathbf{x}_{t-1},\mathbf{u}_{t-1})B(\mathbf{x}_{t-1})}. \tag{13.8}$$

Reward function: The actual error which results from action $\mathbf{u}_t$ is

$$\mathbf{E}_t = \mathbf{y}_t - T \tag{13.9}$$

Average bias: The optimal value of actual error after t run with regards to the transition matrix at each run and learning from the previous runs. Since each action is independent, the only factor that influences the total error is the gain of the actions. Therefore, bias is interpreted as the expected total difference between the reward ($\mathbf{E}_t$) and the gain ($G(\mathbf{E}_t)$).

$$bias = \mathrm{E}\left\{\sum_t \mathbf{E}_t - G(\mathbf{E}_t)\right\} \tag{13.10}$$

In practice when $t \to \infty$, the optimal gain is $\frac{1}{t}\mathrm{E}\left\{\sum_t \mathbf{E}_t\right\}$.

Consider $\mathbf{y}_t$, where $T = 0$ in (13.9) as reward function in tth run then the average optimality reward function based on [13] is

$$Q(\mathbf{x}_t, \mathbf{u}_t) = (1 - \eta_t)Q(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}) + \eta_t \times \left[\mathbf{y}_t - G(\mathbf{y}_t) + \min_{\mathbf{u}_t} Q(\mathbf{x}_{t-1}, \mathbf{u}_{t-1})\right], \tag{13.11}$$

---

[3]Regards to notation in the beginning of this section $\mathbf{x}_t$ is equivalent to S_t; $\mathbf{u}_t$ is A_t; $\mathbf{y}_t$ is O_t; $P(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_{t-1})$ is $R(S_t|S_{t-1}, A_{t-1})$; and E_t is $R(S_t, A_t)$.

where $G(\mathbf{y}_t)$ can be learned and updated at each run by

$$G(\mathbf{y}_t) = (1 - \eta_t')G(\mathbf{y}_{t-1}) + \eta_t' \left[\frac{(t-1)G(\mathbf{y}_{t-1}) + \mathbf{y}_t}{t} \right] \qquad (13.12)$$

The learning parameters η_t and η_t' (similar to λ in EWMA controller) are both decayed at run t by the following rule:

$$\eta_t, \eta_t' = \frac{\eta_0, \eta_0'}{1 + \frac{t^2}{K+t}} \qquad (13.13)$$

where K is a very large number, and η_0, η_0' are initial values for learning parameter η and η', respectively.

Choosing the optimal action: The objective function of a control system in (13.9) can be minimize by the optimal solution of stationary policy given by the observation space:

$$\pi(B(\mathbf{x}_t)) = \arg \min_{r_t} \left[r_t B(\mathbf{x}_t) \times Q(\mathbf{x}_t, \mathbf{u}_t) r_t' \right]. \qquad (13.14)$$

Using the model as a controller: For having a controllable and observable system, the following assumptions should be satisfied:

- The model applies over an infinite number of runs, implying that the control system is stationary.
- Conditioned on the true $\mathbf{u}_t$ and control setting at run t, the $P(\mathbf{y}_t|\mathbf{x}_t)$ is independent from information related to the run $t-1$.
- The measurement noise and process disturbance are accumulated to the $\mathbf{y}_t$ and $\mathbf{u}_t$, respectively.
- Regards to policy function in (13.8), objective function of optimization goal in (13.9) is updated by

$$\arg \min_{r_t} \sum_t \{r_t B(\mathbf{x}_t) \times Q(\mathbf{x}_t, \mathbf{u}_t) r_t\} \qquad (13.15)$$

For investigating the efficiency of the POMDP controller in comparison with EWMA as the most common control filter in the semiconductor industry, an SISO process with 200 runs simulated as follows:

- generating 200 runs of uncontrollable disturbance d_t and noise ε_t from $N(0, 1)$ based on the model in (13.4), where y_1, u_1, x_1 and T are set to zero.
- fitting simulated overlay errors from 200 runs into (13.4) and obtaining the effect of cumulative disturbance on y_t, u_t, and x_t for run $t = 2, \ldots, 200$ where the coefficient parameters (C_1, C_2, C_3) for simplicity set to one.
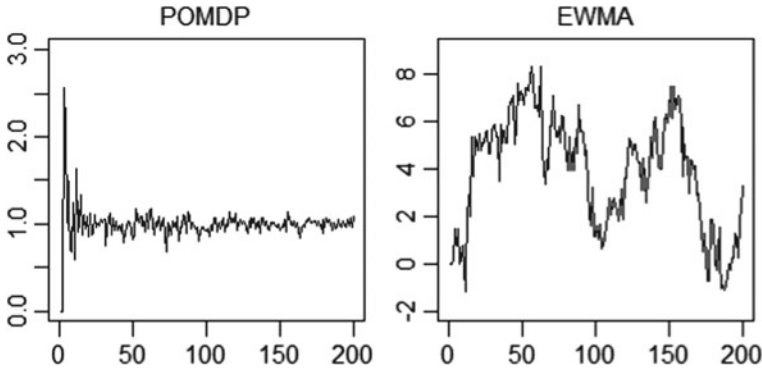
Fig. 13.9 The power of disturbance rejection between POMDP and EWMA controller. Y axis denotes to the total overlay error and X axis to the simulation runs

- calculate the probability distribution function in (13.7) based on empirical distribution function (in reality, we can use historical data to find the distribution function of $\mathbf{x}_t$).
- optimize objective function in (13.14) subject to model in (13.4) for each run where both η_0, η_0' are set to 0.5, and $K = 10^{15}$.
- evaluating the performance of the proposed POMDP controller with 200 runs in comparison with EWMA controller with fixed discount factor equal to 0.3 and $b = 0.5$ as presented in Eqs. (13.1)–(13.3) with $\varepsilon_t = 0$ (note that the effect of noise and disturbance is considered in steps 1 and 2).

Figure 13.9 illustrates this comparison based on the value of (13.9) and shows that how POMDP is performing supremely better than EWMA to compensate the disturbance.

13.6 Conclusion

This study highlighted the importance of disturbance rejection algorithm in semiconductor manufacturing for overlay error minimization during the photolithography process. The research summarized several disturbance rejection algorithms as a comprehensive collection for researchers and practitioners who would like to investigate in this field. However, the algorithm and methods for disturbance rejection are not limited to those are mentioned in this paper. Practically, the hybrid algorithm and technology-enabled method is more efficient than traditional control theory and are more applicable in the smart manufacturing environment.

References

1. Åström, K.J., Wittenmark, B.: Adaptive Control. Courier Corporation (2013)
2. Bode, C., Ko, B., Edgar, T.: Run-to-run control and performance monitoring of overlay in semiconductor manufacturing. Control Eng. Pract. **12**(7), 893–900 (2004)
3. Chen, Y., Moore, K.L., Ahn, H.S.: Iterative learning control. In: Encyclopedia of the Sciences of Learning. Springer, pp. 1648–1652 (2012)
4. Chiang, Y.C., Cheng, C.C.: Terminal adaptive output feedback variable structure control. In: IET Control Theory & Applications (2018)
5. Chien, C.F., Hsu, C.Y.: Unison analysis to model and reduce step-and-scan overlay errors for semiconductor manufacturing. J. Intell. Manuf. **22**(3), 399–412 (2011)
6. Chien, C.F., Chang, K.H., Chen, C.P.: Design of a sampling strategy for measuring and compensating for overlay errors in semiconductor manufacturing. Int. J. Prod. Res. **41**(11), 2547–2561 (2003)
7. Chien, C.F., Chang, K.H., Chen, C.P., Lin, S.L.: Overlay error model, sampling strategy and associated equipment for implementation. US Patent 6,975,974 (2005)
8. Chien, C.F., Chen, Y.J., Hsu, C.Y., Wang, H.K.: Overlay error compensation using advanced process control with dynamically adjusted proportional-integral R2R controller. IEEE Tran. Autom. Sci. Eng. **11**(2), 473–484 (2014)
9. Da Silva, F.D.C., De Oliveira, J.B., De Araujo, A.D.: Robust interval adaptive pole-placement controller based on variable structure systems theory. In: 2017 25th International Conference on Systems Engineering (ICSEng). IEEE, pp. 45–54 (2017)
10. Duarte-Mermoud, M.A., Aguila-Camacho, N., Gallegos, J.A., Travieso-Torres, J.C.: Fractional-order model reference adaptive controllers for first-order integer plants. In: New Perspectives and Applications of Modern Control Theory. Springer, pp. 121–151 (2018)
11. El Chemali, C., Freudenberg, J., Hankinson, M., Collison, W., Ni, T.: Critical dimension control of a plasma etch process by integrating feedforward and feedback run-to-run control. J. Vac. Sci. Technol. B: Microelectron. Nanometer Struct. Process. Meas. Phenom. **21**(6), 2304–2312 (2003)
12. Gamez, D.: Progress in machine consciousness. Conscious. Cogn. **17**(3), 887–910 (2008)
13. Ganesan, R., Das, T.K., Ramachandran, K.M.: A multiresolution analysis-assisted reinforcement learning approach to run-by-run control. IEEE Trans. Autom. Sci. Eng. **4**(2), 182–193 (2007)
14. Ge, S.S., Hang, C.C., Lee, T.H., Zhang, T.: Stable Adaptive Neural Network Control, vol 13. Springer Science & Business Media (2013)
15. Giarratano, J.C., Riley, G.: Expert Systems: Principles and Programming. Brooks/Cole Publishing Co. (1989)
16. Gong, Q., Yang, G., Pan, C., Chen, Y.: Performance analysis of single ewma controller subject to metrology delay under dynamic models. IISE Trans. **50**(2), 88–98 (2018)
17. Gong, Q.S., Yang, G.K., Pan, C.C., Lee, M.S.: Stability and control performance analysis of double ewma controller with metrology delay. IEEE Trans. Semicond. Manuf. **29**(1), 9–16 (2016)
18. Good, R.P., Qin, S.J.: On the stability of MIMO EWMA run-to-run controllers with metrology delay. IEEE Trans. Semicond. Manuf. **19**(1), 78–86 (2006)
19. Gross, E.: On the Bellman's principle of optimality. Phys. A: Stat. Mech. Appl. **462**, 217–221 (2016)
20. Guo, B.Z., Zhao, Z.L.: On convergence of tracking differentiator. Int. J. Control **84**(4), 693–701 (2011)
21. Guo, B.Z., Zhao, Z.L.: Extended state observer. In: Active Disturbance Rejection Control for Nonlinear Systems: An Introduction, pp. 93–154 (2016)
22. Haddad, W.M., Chellaboina, V.: Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach. Princeton University Press (2011)
23. van Hee, K.M., Hee, K.: Bayesian Control of Markov Chains, vol. 95. Mathematisch centrum Amsterdam, The Netherlands (1978)

24. Heirung, T.A.N., Ydstie, B.E., Foss, B.: Dual adaptive model predictive control. Automatica **80**, 340–348 (2017)

25. Huang, M., Wang, X., Lu, Z., Ma, L., Su, H., Wang, L.: Multiple model adaptive control for a class of nonlinear systems with unknown control directions. Int. J. Control 1–13 (2018)

26. Huang, Y., Xue, W.: Active disturbance rejection control: methodology and theoretical analysis. ISA Trans. **53**(4), 963–976 (2014)

27. Jang, J.S., Sun, C.T.: Neuro-fuzzy modeling and control. Proc. IEEE **83**(3), 378–406 (1995)

28. Khakifirooz, M., Chien, C.F., Chen, Y.J.: Bayesian inference for mining semiconductor manufacturing big data for yield enhancement and smart production to empower industry 4.0. Appl. Soft Comput. **68**, 990–999 (2018)

29. Khakifirooz, M., Fathi, M., Chien, C.F.: Modelling and decision support system for intelligent manufacturing: an empirical study for feedforward-feedback learning-based run-to-run controller for semiconductor dry-etching process. Int. J. Ind. Eng. Theory Appl. Pract. **25**(6) (2018)

30. Khakifirooz, M., Chien, C.F., Fathi, M.: Compensating misalignment using dynamic random-effect control system: a case of high-mixed wafer fabrication. IEEE Trans. Autom. Sci. Eng. (2019)

31. Khakifirooz, M., Cayard, D., Chien, C.F., Fathi, M.: A system dynamic model for implementation of industry 4.0. In: 2018 International Conference on System Science and Engineering (ICSSE). IEEE, pp. 1–6, June 2018

32. Kuo, H.F., Faricha, A.: Artificial neural network for diffraction based overlay measurement. IEEE Access **4**, 7479–7486 (2016)

33. Liptk, B.: Process Control: Instrument Engineers Handbook (2013)

34. Lynn, S.A., MacGearailt, N., Ringwood, J.V.: Real-time virtual metrology and control for plasma etch. J. Process Control **22**(4), 666–676 (2012)

35. Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press (1998)

36. Monahan, G.E.: State of the art-a survey of partially observable Markov decision processes: theory, models, and algorithms. Manag. Sci. **28**(1), 1–16 (1982)

37. Moreau, W.M.: Semiconductor Lithography: Principles, Practices, and Materials. Springer Science & Business Media (2012)

38. Moyne, J., Del Castillo, E., Hurwitz, A.M.: Run-to-Run Control in Semiconductor Manufacturing. CRC Press (2000)

39. Ortega, R., Perez, J.A.L., Nicklasson, P.J., Sira-Ramirez, H.J.: Passivity-Based Control of Euler-Lagrange Systems: Mechanical, Electrical and Electromechanical Applications. Springer Science & Business Media (2013)

40. Park, S.J., Lee, M.S., Shin, S.Y., Cho, K.H., Lim, J.T., Cho, B.S., Jei, Y.H., Kim, M.K., Park, C.H.: Run-to-run overlay control of steppers in semiconductor manufacturing systems based on history data analysis and neural network modeling. IEEE Trans. Semicond. Manuf. **18**(4), 605–613 (2005)

41. Rico-Azagra, J., Gil-Martínez, M., Rico, R., Maisterra, P.: Qft bounds for robust stability specifications defined on the open-loop function. Int. J. Robust Nonlinear Control **28**(3), 1116–1125 (2018)

42. Rugh, W.J., Shamma, J.S.: Research on gain scheduling. Automatica **36**(10), 1401–1425 (2000)

43. Schreier, M.: Modeling and adaptive control of a quadrotor. In: 2012 International Conference on Mechatronics and Automation (ICMA). IEEE, pp. 383–390 (2012)

44. Sullivan, N.T.: Semiconductor pattern overlay. In: Handbook of Critical Dimension Metrology and Process Control: A Critical Review. International Society for Optics and Photonics, vol. 10274, p. 102740C (1994)

45. Sun, L., Lu, J., Liu, Y., Huang, T., Alsaadi, F.E., Hayat, T.: Variable structure controller design for Boolean networks. Neural Netw. **97**, 107–115 (2018)

46. Tan, Y., Moase, W., Manzie, C., Nešić, D., Mareels, I.: Extremum seeking from 1922 to 2010. In: 2010 29th Chinese Control Conference (CCC). IEEE, pp. 14–26 (2010)

47. Tanaka, T., Esfahani, P.M., Mitter, S.K.: LQG control with minimum directed information: semidefinite programming approach. IEEE Trans. Autom. Control **63**(1), 37–52 (2018)

48. Thie, P.R.: Markov Decision Processes. Comap, Incorporated (1983)
49. Utkin, V., Guldner, J., Shi, J.: Sliding Mode Control in Electro-Mechanical Systems. CRC Press (2009)
50. Vassilyev, S., Kelina, A.Y., Kudinov, Y., Pashchenko, F.: Intelligent control systems. Procedia Comput. Sci. **103**, 623–628 (2017)
51. Wan, L., Tan, F., Th, P.A.N.: Online estimation of time-varying metrology delay and run-to-run control co-design. Control Theory Appl. **1**, 012 (2016)
52. Wang, Y., Zheng, Y., Fang, H., Wang, Y.: Armax model based run-to-run fault diagnosis approach for batch manufacturing process with metrology delay. Int. J. Prod. Res. **52**(10), 2915–2930 (2014)
53. Wu, W.M., Cheng, F.T., Lin, T.H., Zeng, D.L., Chen, J.F.: Selection schemes of dual virtual-metrology outputs for enhancing prediction accuracy. IEEE Trans. Autom. Sci. Eng. **8**(2), 311–318 (2011)
54. Xie, J., Yang, D., Zhao, J.: Multiple model adaptive control for switched linear systems: a two-layer switching strategy. Int. J. Robust Nonlinear Control **28**(6), 2276–2297 (2018)
55. Zheng, Y., Wong, D.S.H., Wang, Y.W., Fang, H.: Takagi-Sugeno model based analysis of EWMA RtR control of batch processes with stochastic metrology delay and mixed products. IEEE Trans. Cybern. **44**(7), 1155–1168 (2014)
56. Zimmermann, H.J.: Fuzzy control. In: Fuzzy Set Theory-and Its Applications. Springer, pp. 203–240 (1996)

Chapter 14
Energy Management Improvement Based on Fleet Digitalization Data Exploitation for Hybrid Electric Buses

Jon Ander López, Victor Isaac Herrera, Haritza Camblong, Aitor Milo and Haizea Gaztañaga

Abstract The chapter focuses on a fleet energy management approach with the aim of reducing operation and maintenance costs. A state-of-the-art is presented for the different proposed fleet management approaches. In order to tackle the digitalization challenge of exploiting the large data volume of a fleet of vehicles, a methodology for improving electrified buses energy efficiency at fleet level is proposed. In addition, an energetic analysis of a fleet based on this methodology has been performed. The analyzed fleet is composed of buses with parallel and series configurations and include energy storage systems based on batteries and ultracapacitors. In the first stage, a dynamic programming approach has been applied to determine the initial optimal operation performance for each bus route. Then, several disruptions (e.g., traffic jams, auxiliary consumption, and passenger variations) have been added to the routes to simulate "real" road and daily operation conditions. This data is used for monitoring the energetic key performance factors by learning from the buses with the best energetic behavior. Finally, a decision-making process is applied to improve the local energy management of the less-efficient bus.

J. A. López (✉) · A. Milo · H. Gaztañaga
IKERLAN Technology Research Centre, Energy Storage and Management Area,
Arrasate, Gipuzkoa, Spain
e-mail: jonander.lopez@ikerlan.es

V. I. Herrera
Facultad de Informática y Electrónica, Escuela Superior Politécnica de Chimborazo,
Riobamba, Ecuador
e-mail: isaac.herrera@espoch.edu.ec

H. Camblong
University of the Basque Country, Donostia San Sebastian, Gipuzkoa, Spain

ESTIA Research, Bidart, France
e-mail: aritza.camblong@ehu.eus

14.1   Introduction

In the last few years, the reduction of greenhouse gas emissions has become a major concern. This growing concern has been driven by the current state of the road transport. In 2016, this sector caused the 20% of the total greenhouse gas (GHG) in Europe. In terms of road transport GHG emission reduction, light-duty vehicles have been in the spotlight, almost neglecting the heavy-duty vehicles (HDV)s impact. HDVs emit the quarter part of the whole road transport GHG emissions, despite the established more restrictive EURO VI GHG emissions standard, which limits the emissions nearly to zero. Moreover, due to the high pollution concentration in some cities, zero-emission zones have been implemented, as shown in Fig. 14.1 [1] with a tendency to increase over the years.

It is highlighted that buses and lorries do not affect in the same level. The most polluted zones are urban areas. Freight transport is much less concentrated in urban areas than the urban public transport, as they are commonly used in longer transport distances. However, public transport is mostly running in urban areas around 16 hours a day. Adding to this fact, buses are the most used means of public transport. Therefore, buses are an important candidate for electrification.

Due to the low maturity level, the initial investment for hybrid and electric buses is significantly higher than in the case of conventional (fossil fuel-based) buses. On the contrary, several studies reveal that, due to lower operation costs throughout the lifetime of the vehicle in some particular situations (countries with high fuel prices), the total cost of ownership is lower in the case of hybrid and electric buses beside conventional buses.
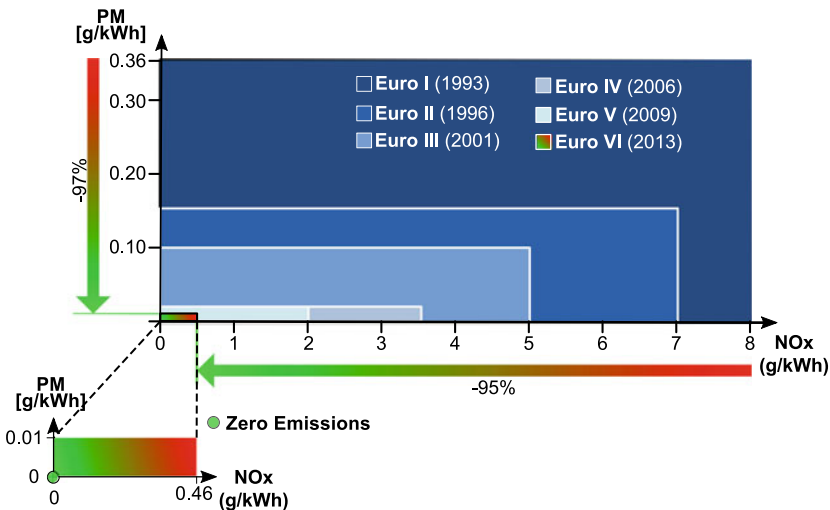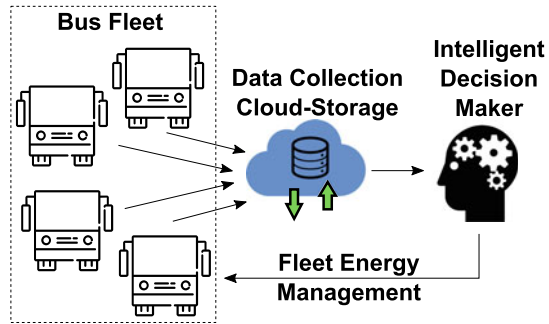

Fig. 14.1   EURO standard greenhouse gas emission limitations

Fig. 14.2 Digitalization of urban transport opportunities

In the automotive industry, there is a new trend for digitalizing vehicles. The digitalization of vehicles is the process of providing vehicles with sensors to acquire information, storing this information in the cloud data storage, and analyzing the data. This new trend enables to monitor the energetic operation of vehicles widening the opportunities to improve the energetic efficiency of the vehicles, as depicted in Fig. 14.2. However, the main challenge for the digitalization is the large data volume to be managed by fleet managers. Therefore, new automated tools are needed, in order to analyze, process, and make decisions based on this processed data.

The current approach for improving the energy efficiency is based on applying local energy management strategies (EMSs), which are optimized for each vehicle and route. Commonly, this strategy is neither updated throughout the vehicle lifetime nor compared with the energetic performance of other vehicles in similar operation scenarios. Here, the main issue is the lack of data for re-evaluating the strategy or the lack of intelligent tools to analyze multiple cases (vehicles on routes) to find improved ways to manage the energy and operate the local systems more efficiently. The aforementioned new trend of digitalization changes the traditional scenario. A new level is identified above the energy management at local vehicle level, with the aim of managing the whole fleet. This new level opens up a new scope in energy management, allowing to compare different vehicles' energetic behavior, learning from those with the best energetic performance.

The new digitalization tendency with the intelligent tools will allow to monitor the whole fleet operation, improving the fleet energetic efficiency, and reducing operation costs. As a result, the total cost of ownership will be controlled, with the aim to improve as much as possible, making hybrid and electric vehicles more competitive methodology for the analysis and management of data from a fleet of vehicles. Consequently, the contribution of this paper lies on a proposed methodology for managing the data at fleet level, increasing the overall fleet energy efficiency.

This paper is an extended version of [2] and it is organized into six sections. The first section presents the state-of-the-art of the different proposed fleet management approaches in the literature. In the second section, the analyzed fleet case study is presented. The third section describes the basic theoretical concepts. First, the electrical modeling of the series and parallel HEB configurations are explained. After

that, the optimization technique is introduced. Finally, the economic model for the fleet lifetime operation cost estimation is presented. The fourth section corresponds to the proposed fleet learning methodology. In the fifth section, the obtained results and analysis are presented. In the sixth and last section, the general conclusions and contributions extracted from this chapter are presented.

14.2 Fleet Management Approaches

All sectors in Europe have recorded a GHG emission reduction since 1990, excepting the transport sector [3], being the road transport responsible for the 72.1% of total transport GHG emissions in 2016, as depicted in Fig. 14.3A [4]. Despite the more restrictive established standards regarding GHG emissions (EURO VI) limiting the Internal Combustion Engine (ICE) operation, HDVs emissions have been constantly growing between 1990 and 2010, Fig. 14.3B [3]. The European bus fleets layout, still in 2015 was composed of nearly 50% of the vehicles with Euro III or older ICEs [5]. Moreover, nowadays about 98% of the lorries in Europe rely on diesel [6]. In order to fulfill the scheduled reduction of 30% of CO_2 emissions for HDVs for the year 2030, from 2019 levels [6], a renewal of the road transport fleets to a more sustainable scenario is needed.

Analyzing the level of pollution impact of lorries and buses is different. The most worrying zones are urban areas, due to the high pollution and congestion levels, where commonly buses are running around 16 h a day. Moreover, buses are the most used type of public transport, with nearly 56% of journeys in Europe in 2014, as shown in Fig. 14.3C [7]. Therefore, with the aim of reducing urban areas pollution levels, hybrid and electric buses need to integrate. Furthermore, several studies have been carried out in terms of analyzing the alternative road transport HDVs different solutions. The main alternative available technologies are biofuels, gas, fuel cell, hybrid, and full-electric-based vehicles. The conclusion of this study states that both hybrid and full-electric buses' solutions are the most viable ones [8–15].

Despite being the feasible solutions, hybrid and electric buses' integration is a challenging process. Several studies pointed out the high initial investment cost
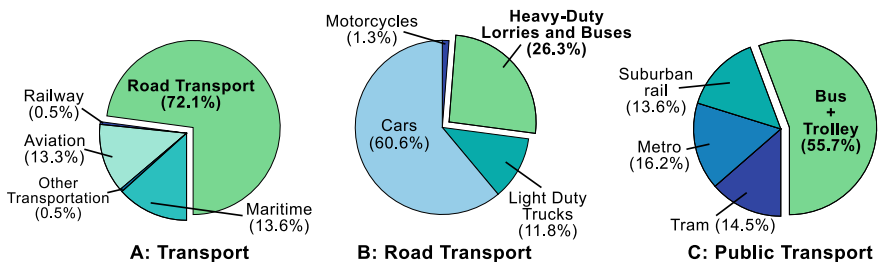

Fig. 14.3 Public urban transport breakdown

besides conventional buses [9, 16–20]. To overcome this problem and to make a more attractive transport for investors, the total cost of ownership (TCO) is the key point to focus on. TCO is the economic performance indicator, which includes manufactured price and the costs for maintenance, operation, energy distribution, infrastructure, emission, insurance, and end-of-life [18]. In this regard, several studies revealed that operation and maintenance costs in the case of hybrid and electric HDVs throughout the whole lifetime can be lower than the diesel buses [9, 13, 18, 21–23].

The TCO calculation is highly dependent to the operational aspect [18]. In [24], it was studied an intention to replicate a full-electric bus fleet operation in real conditions. The outcome of this research study was to obtain different energy consumptions and performances among the bus fleet. As a result, there exit uncertainties in the literature for the TCO estimation, regarding the real operation conditions [18]. Therefore, with the aim of reducing uncertainties, there is a need to monitor the vehicles' operation [25].

The continuous development of smart devices and the implementation of telematics has resulted in a new information source. Traditionally, telematics have been used for vehicle positioning. However, the new trend of cloud data storage and data analysis, known as digitalization process, have derived in new services [26, 27]. The newly provided services are oriented to energy savings, driving behavior, dynamic routing, charging schedule, and diagnostics [26, 28].

In the literature, traditionally the energy management has been composed of three levels [29, 30], as shown in Fig. 14.4. In this traditional approach, the EMS plays a key role in order to manage the energetic behavior of each single vehicle. Therefore, transport local EMSs have been a topic of interest for several years, with the aim of improving the energy efficiency of each single vehicle [31–33].

The traditional EMS approach evaluates at single vehicle level, having a relatively limited scope [33]. A classification of the EMSs is depicted in Fig. 14.5 [30]. However, the digitalization trend has allowed to go a step further in the energy
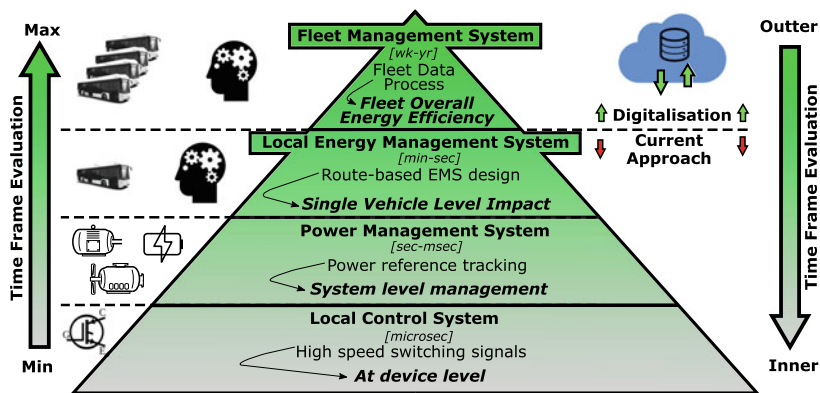

Fig. 14.4 Representation of the hierarchical management and control of a power electronic system, with the proposed fleet management system
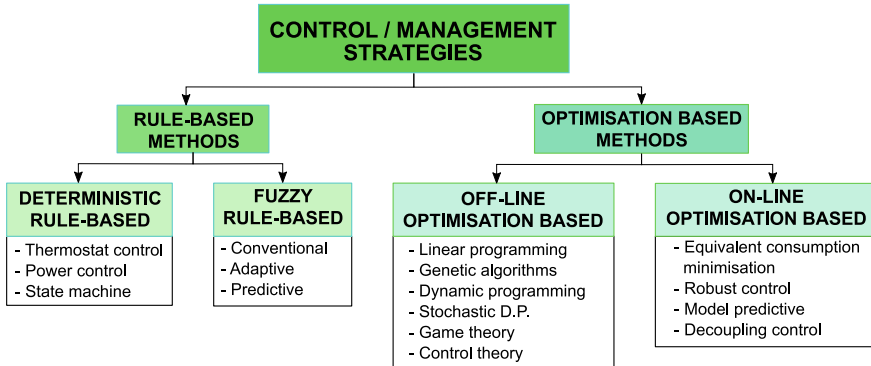
Fig. 14.5 Energy management approaches for HEBs

management hierarchy, adding an additional level shown in Fig. 14.4. Collecting and processing information that affects energy consumption at fleet level enables to widen the EMS scope, becoming a thriving area of research [28, 33].

The main objective of the fleet energy management system is to analyze, process, and make decisions based on the processed data. A wider scope allows to compare the behavior of each vehicle on the fleet, learning from those with the best energetic behavior. Thus, the aim is to improve the whole fleet energy efficiency, reducing the operation and maintenance costs.

In the literature, several fleet management system approaches have been identified, with different purposes [32, 33]. Examining the proposed approaches, for a better understanding a classification ,has been done, based on the topics of each fleet management system approach. The main identified purposes are focused on the following areas: traffic jams avoidance, vehicle diagnostics, itinerary planning, and charging regulation and scheduling. In the following lines is done an overview and a literature review with regard to the aforementioned classification.

Traffic jams avoidance has been the first developed fleet management system. As it has been aforementioned, the first use of telematics was focused on vehicle positioning. In this regard, Thong et al. proposed a fleet management system for location accuracy improvement, with the aim of developing an intelligent system for avoiding traffic jams [34]. In addition, in [35] Balaji et al. aimed to develop a smart traffic light time optimization approach, to avoid traffic jams. Likewise, HomChaudhuri et al. proposed a hierarchical control for energy management with two levels of sharing information. The higher level is based on the traffic lights information and traffic information (provided by the surrounding vehicles). This information is used for predicting the velocity and choosing the target velocity with the best fuel efficiency. The lower level with the above level information controls the power split factor of the hybrid vehicle by applying the energy consumption minimization strategy [36].

In the automotive industry, *vehicle diagnostics* for predictive maintenance has been the driving force in digitalization implementation. J. Grantner et al. proposed an intelligent vehicle health diagnostics method, based on fuzzy logic, updating the fuzzy rules from cluster extraction of the fleet data [37]. In regard to the fleet battery (BT) data exploitation, the following approaches have been proposed. Haycock et al. proposed a BT monitoring approach, with the aim to evaluate the BT utilization and improve the TCO [38]. Likewise, Barré et al. approach a methodology for clustering EVs according to each operation, with the aim to improve the BT state of health (SOH) estimation [39].

Another topic of research, based on data exploitation, is the *green itinerary planning*. In the literature, several approaches have been proposed [40–43]. It is worth mentioning the approach of Mehar et al. proposing an extension of the EcoDrive green itinerary planning service [42] for EVs providing several paths taking into account the following information. (i) Road cartography: elevation; (ii) Driving perturbations: traffic congestions, unexpected events, and driver habits; (iii) Vehicle features: weight, BT type, and engine efficiency map; and (iv) Weather conditions: air friction, wind speed, and temperature.

Recharging infrastructure optimization at fleet level has been topic of research for HDVs, particularly applied to urban mobility, due to the fact that the profiles are predefined. The compilation and processing of information of the recharging operation widen the recharging management opportunities. Qin et al. exploited the compiled information with the aim of minimizing the charging station queues [44]. In an attempt to go a step further on recharging management, Hill et al. proposed a vehicle-to-grid (V2G) fleet model. This model shows that with BT operation monitoring, the shallow cycles are avoided. Consequently, the BT lifetime increases, making more profitable the V2G at fleet level. Lastly, Rogge et al. [45] address the challenges of range limitation and required charging time, developing a methodology for the charging infrastructure planning.

In regard to fleet data exploitation in [46], Wittmann et al. proposed a holistic framework. This approach covers the fleet tracks data acquisition to the evaluation. The data is acquired from a smartphone application and a data logger. Consequently, this data is processed filtering the tracks, ordering according to the selected features (such as distance, average speed, and driving behavior). Finally, the data is evaluated making use of different developed tools. It is noteworthy the developed data analysis tool extension used to simulate electric vehicle's energy consumption. However, this approach is not focused on a methodology for fleet energy management.

After the literature review, it can be concluded that there is a lack of methodologies, strategies, and tools for the energetic behavior improvement at fleet level. Therefore, a gap in the literature is identified, for developing tools to manage fleet management systems, in regard to fleet energetic efficiency improvement to exploit the upcoming opportunities in this thriving area of research [28, 33].

14.3 Case Study

Based on the proposed methodology, an energetic analysis of a fleet has been performed. Some data of the scenario analyzed in this paper is approached in Table 14.1. It focuses on three different HEB configurations as shown in Fig. 14.6:

- *Parallel HEB with Battery (Par)*: a power train pulled with an ICE and an electric motor (EM), operated by a BT pack.
- *Series HEB with Battery (Ser 1)*: a power train pulled by an EM powered by a BT pack and a genset (GS).
- *Series HEB with Ultracapacitor (Ser 2)*: a power train pulled by an EM powered by an ultracapacitors (UCs) pack and a GS.

The main parameters of the used BT and UC cells are summarized in Table 14.2. It is noteworthy that the BT cell has been selected due to the high c-rates, which allows fast charging and longer BT lifetime than other lithium-based chemistries.

The urban routes have been generated from a database of standardized driving cycles generating new profiles from two profiles' combination and round trips, as

Table 14.1 Scenario approach

	Par	Ser 1	Ser 2
Driving cycle profiles/bus configuration	14	4	14
Electric motor power (kW)	220	220	220
Internal combustion engine power (kW)	170	–	–
Genset power (kW)	–	170	170
Battery pack energy (kWh)	24	24	–
Ultracapacitor energy (kWh)	–	–	1
Energy storage system weight (kg)	266	266	145.8
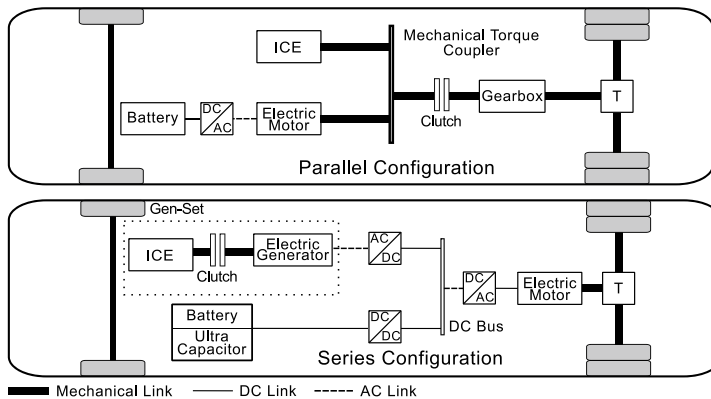
Fig. 14.6 Parallel and series HEB configurations

Table 14.2 Electrical parameters of BT- and UC-based cells

Battery		Ultracapacitor	
Nom. voltage	2.3 V	Nom. voltage	2.7 V
Nom. capacity	20 Ah	Nom. capacitance	3000 F
Int. resistance	0.53 mΩ	Int. resistance	0.29 mΩ
Max C-rate disch/ch	7/8 C-rate	Max C-rate disch/ch	110/110 C-rate
Specific energy	90 Wh/kg	Specific energy	6.0 Wh/kg
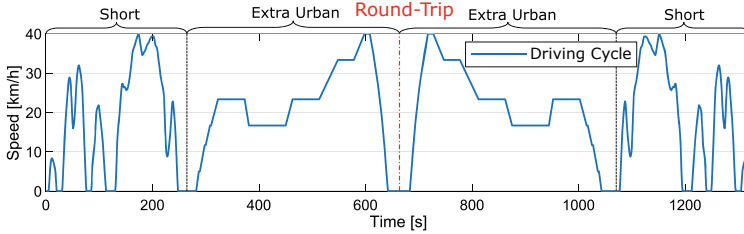
Fig. 14.7 Standardized cycles' combination and round-trip generation

depicted in Fig. 14.7. Each of these generated cycles has been applied to an HEB, as explained in Table 14.1, generating a simulation of the fleet.

14.4 System Modeling and Optimization Methodology

In this section, the used HEB electrical models and the optimization technique for the EMS development are described. These HEB models have been implemented in MATLAB, in order to simulate and energetically evaluate the proposed methodology. Finally, an economic model is introduced, used for the lifetime operation cost calculation.

14.4.1 Electrical Model

The electrical model of the power train elements has been developed for the quasi-static simulation method. Therefore, the used formulation has been based on backward or "effect–cause" approach. The power is calculated at each discrete step following a predefined speed profile going upstream through the vehicle components [47]. In order to standardize the power flow direction, the adopted sign convention has been positive power when there is an electrical power demand or mechanical traction and negative when there is an electrical power absorption or mechanical braking.

14.4.1.1 Bus Dynamics

In the quasi-static simulation, the inputs to the vehicle model are the speed $v_{cyc}(k)(\frac{m}{s})$, acceleration $a_{cyc}(k)$ $(\frac{m}{s^2})$ and the slope angle $\alpha(k)$ (°) of the predefined route [47]. From these profiles, the backward simulation is applied, starting from the calculation of the force acting on the wheels (F_T), at each discrete state k defined as follows [30, 47]:

$$F_T(k) = F_a(k) + F_g(k) + F_i(k) + F_r(k) \quad \text{(N)} \tag{14.1}$$

being $F_a(k)$ (N) aerodynamic force, $F_g(k)$ (N) gravitational force, $F_i(k)$ (N) inertial force, and $F_r(k)$ (N) rolling resistance force (depicted in Fig. 14.8), at each discrete state k are defined as follows:

$$F_a(k) = 0.5 \cdot \rho_{air} \cdot A_f \cdot c_x \cdot v_{cyc}^2(k) \quad \text{(N)} \tag{14.2}$$

$$F_g(k) = m_{tot} \cdot g \cdot sin(\alpha(k)) \quad \text{(N)} \tag{14.3}$$

$$F_i(k) = m_{tot} \cdot a_{cyc}(k) \quad \text{(N)} \tag{14.4}$$

$$F_r(k) = c_{rf} \cdot m_{tot} \cdot g \cdot cos(\alpha(k)) \quad \text{(N)} \tag{14.5}$$

being ρ_{air} $\left(\frac{kg}{m^3}\right)$ the density of air, A_f (m^2) the frontal area of the vehicle, c_x (–) the aerodynamic drag, m_{tot} (kg) total vehicle mass, g $\left(\frac{m}{s^2}\right)$ the gravitational acceleration, and c_{rf} (–) the road rolling coefficient [30].

The total mass of the vehicle can be defined as

$$m_{tot} = m_{veh} + m_{ESS} + m_{pass} \cdot n_{pass} \quad \text{(kg)} \tag{14.6}$$

where m_{veh} (kg) is the empty bus weight, m_{ESS} (kg) is the ESS weight, m_{pass} (kg) is the average weight per person (assumed to be 75 kg), and n_{pass} is the number of passengers [30].

Fig. 14.8 Forces acting on the bus during driving
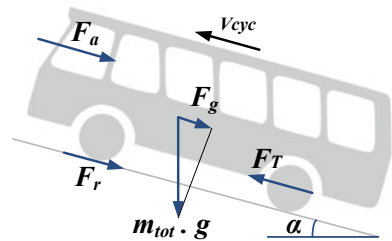
From the bus dynamic model calculation, the outputs are the wheel rotational speed $w_{wh}(k)$ $\left(\frac{rad}{s}\right)$, acceleration $dw_{wh}(k)$ $\left(\frac{rad}{s^2}\right)$, and the required torque in the wheel $T_{wh}(k)$ (Nm) calculated as follows:

$$w_{wh}(k) = \frac{v_{cyc}(k)}{r_{wh}} \quad \left(\frac{rad}{s}\right) \tag{14.7}$$

$$dw_{wh}(k) = \frac{a_{cyc}(k)}{r_{wh}} \quad \left(\frac{rad}{s^2}\right) \tag{14.8}$$

$$T_{wh}(k) = F_T(k) \cdot r_{wh} \quad (Nm) \tag{14.9}$$

where $v_{cyc}(k)\left(\frac{m}{s}\right)$ is the cycle speed, $a_{cyc}(k)$ $\left(\frac{m}{s^2}\right)$ is the cycle acceleration, and r_{wh} (m) is the wheel radius.

14.4.1.2 Transmission Model

The transmission consists of the elements placed between the motor and the drive wheel axle. For the case of the series configuration, as the EM is more flexible in a wider rotational speed ranges, there is no need of a gearbox and it has directly a transmission. In the case of the parallel configuration, there is an additional element in the power train, the gearbox, as depicted in Fig. 14.6. The gearbox transforms a certain rotational speed to a different speed, with the aim of making the most of the ICE [47].

In both cases, the inputs are the outputs of the dynamic model $w_{wh}(k)$, $a_{wh}(k)$, and $T_{wh}(k)$. As a result, the rotational speed of the drive shaft $w_{drsft}(k)$ $\left(\frac{rad}{s}\right)$, acceleration of the drive shaft $dw_{drsft}(k)$ $\left(\frac{rad}{s^2}\right)$, and the required torque in the drive shaft $T_{drsft}(k)$ (Nm) are recalculated as follows:

$$w_{drsft}(k) = w_{wh}(k) \cdot \gamma(k) \quad \left(\frac{rad}{s}\right) \tag{14.10}$$

$$dw_{drsft}(k) = dw_{wh}(k) \cdot \gamma(k) \quad \left(\frac{rad}{s^2}\right) \tag{14.11}$$

$$T_{drsft}(k) = \frac{T_{wh}(k)[+]}{\gamma(k) \cdot \eta} + \frac{T_{wh}(k)[-] \cdot \eta}{\gamma(k)} \quad (Nm) \tag{14.12}$$

where γ is the gear ratio and η the efficiency of transmission model [47].

The parallel configurations EM and ICE are mechanically coupled; therefore, the required traction is given by the torque demand $T_{dem}(k)$ (N).

$$T_{dem}(k) = T_{drsft}(k) + dw_{drsft}(k) \cdot J_{ICE}(k) + dw_{drsft}(k) \cdot J_{EM}(k) \quad (N) \tag{14.13}$$

where $J_{ICE}(k)$ and $J_{EM}(k)$ are the ICE and EM inertia, respectively.

In the case of the series configuration, the traction is only provided by the EM $T_{EM}(k)$ (Nm). However, the EM power supply is divided between the ESS and a GS, needing to calculate total the power demand $P_{dem}(k)$ (W).

$$T_{EM}(k) = T_{drsft}(k) + dw_{drsft}(k) \cdot J_{EM} \quad \text{(Nm)} \tag{14.14}$$

$$P_{dem}(k) = w_{drsft}(k) \cdot T_{EM}(k) \quad \text{(W)} \tag{14.15}$$

14.4.1.3 Split Factor

The information obtained from the transmission model is used to set the required tractive demand in the backward model. This demand has to be satisfied by each vehicle, combining as energetically efficient as possible the power sources. The combination of the power sources use is determined by the split factor. In regard to each configuration, the variable to split is different.

As it has been aforementioned, for the parallel configuration the torque $T_{dem}(k)$ has to be split. On the contrary, for the series configuration, the power $P_{dem}(k)$ has to be split. In the following lines, a further explanation for each configuration is given.

Parallel HEB
In the parallel configuration, the EM and the ICE are mechanically coupled and rotating at the same speed imposed by the wheels, as shown in Fig. 14.6. The variable deduced by the split factor $U(k)$ is the torque $T_{dem}(k)$ (Nm). The tractive torque demand is divided between the ICE torque $T_{ICE}(k)$ (Nm) and the EM torque $T_{EM}(k)$ (Nm), as shown in Equation (14.16).

$$T_{dem}(k) = \begin{cases} T_{ICE}(k) = T_{dem}(k) \cdot (1 - U(k)) & \text{(Nm)} \\ T_{EM}(k) = T_{dem}(k) \cdot U(k) & \text{(Nm)} \end{cases} \tag{14.16}$$

Series HEB
The series configuration is only driven by the EM. In this case, the split factor represents the power demand $P_{dem}(k)$ (W), as the series configuration is electrically coupled by the electric bus. This factor is divided between the ESS power $P_{ESS}(k)$ (W), BT or UC, and the GS power $P_{GS}(k)$ (W), as shown in Fig. 14.6. The GS is composed of an ICE (speed controlled) and an electric generator (torque controlled).

$$P_{dem}(k) = \begin{cases} P_{GS}(k) = P_{dem}(k) \cdot (1 - U(k)) & \text{(W)} \\ P_{ESS}(k) = P_{dem}(k) \cdot U(k) & \text{(W)} \end{cases} \tag{14.17}$$

14.4.1.4 Electric Motor

As it has been aforementioned, both power train configurations use the EM for traction purposes. The efficiency of the EM $\eta_{EM}(k)$ (%) is calculated by means of the
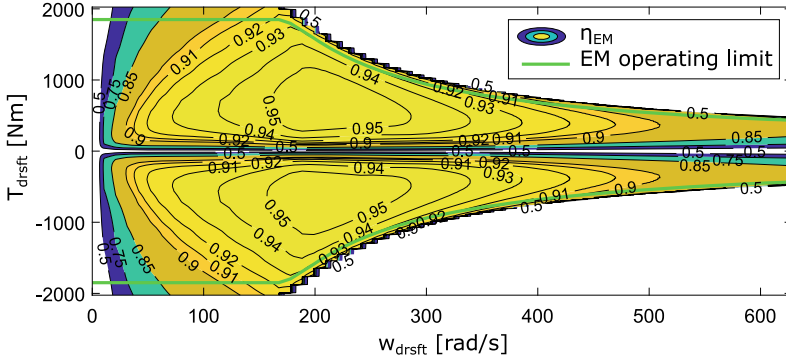
Fig. 14.9 EM efficiency map

$w_{drsft}(k)$ and $T_{EM}(k)$ parameters, based on the efficiency map shown in Fig. 14.9. The EM model output is the required electric power $P_{EM}(k)$ (kW) defined as follows [30]:

When $w_{drsft}(k) > 0$ and $T_{EM}(k) > 0$, (traction mode):

$$P_{EM}(k) = \frac{w_{drsft}(k) \cdot T_{EM}(k)}{10^3 \cdot \eta_{EM}(k)(w_{drsft}(k), \ T_{EM}(k))} \quad \text{(kW)} \tag{14.18}$$

When $w_{drsft}(k) > 0$ and $T_{EM}(k) < 0$, (regenerative mode):

$$P_{EM}(k) = \frac{w_{drsft}(k) \cdot T_{EM}(k) \cdot \eta_{EM}(k)(w_{drsft}(k), \ T_{EM}(k))}{10^3} \quad \text{(kW)} \tag{14.19}$$

14.4.1.5 Internal Combustion Engine

The ICE is used for different purposes depending on the configuration of the power train. On the one hand, for the parallel power train, the ICE is used for traction purpose, mechanically coupled speed coupling [48]. On the other hand, in the case of the series power train configuration, the ICE is used for driving the generator (connected by a clutch), in order to power the EM and the ESSs.

This power target can be obtained from different combinations of ICE rotational speed [$(w_{drsft}(k) \left(\frac{\text{rad}}{\text{s}}\right))$] and torque [$(T_{ICE}(k) \text{ (Nm)})$]. Therefore, the instantaneous fuel mass flow $mf_{ICE}(k) \left(\frac{\text{kg}}{\text{s}}\right)$ consumed at each discrete state k can be defined as follows [30, 47]:

$$mf_{ICE}(k) = f(w_{drsft}(k), \ T_{ICE}(k)) \quad \left(\frac{\text{kg}}{\text{s}}\right) \tag{14.20}$$

The instantaneous fuel mass flow consumed from the ICE is obtained by interpolating the ICE speed and torque value in the fuel consumption map depicted in
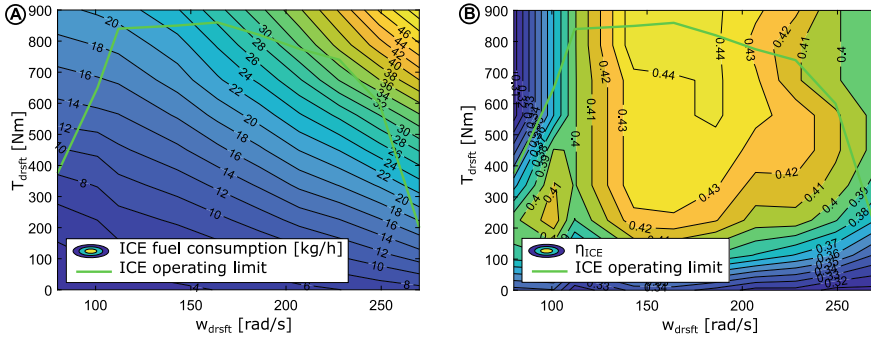
Fig. 14.10 ICE fuel consumption and efficiency maps

Fig. 14.10. In this case, the data of a commercial diesel motor of VOLVO used in hybrid buses applications has been considered for the vehicle modeling.

The power generated by the ICE in the case of parallel configuration is calculated as follows:

$$P_{ICE}(k) = \frac{w_{drsft}(k) \cdot T_{ICE}(k)}{\eta_{ICE}(k)} \quad \text{(kW)} \tag{14.21}$$

where $\eta_{ICE}(k)$ (%) is the ICE efficiency (deducted from the ICE efficiency map depicted in Fig. 14.10B).

On the contrary, for the series configuration, as the ICE drives the generator of the GS, the ICE power can be deduced from Eq. (14.22) [30]:

$$P_{ICE}(k) = \frac{P_{GS}(k)}{\eta_{GS}(k) \cdot \eta_{INV}(k)} \quad \text{(kW)} \tag{14.22}$$

where $\eta_{INV}(k)$ (%) is the average inverter efficiency and $\eta_{GS}(k)$ is the GS efficiency (obtained from the GS efficiency map shown in Fig. 14.11).

14.4.1.6 Genset

The GS is only used for the series power train configuration and is made up of an ICE and an electric generator. The GS model input is the power target $P_{GS}(k)$ (kW) determined by the split factor.

The fuel consumption map has been previously optimized in order to identify the most efficient operation points for the whole power operation range of the GS. Figure 14.11 [30] depicts the optimal operation curve for the GS, over which it will operate. This curve has been included in the simulation model to obtain (depending
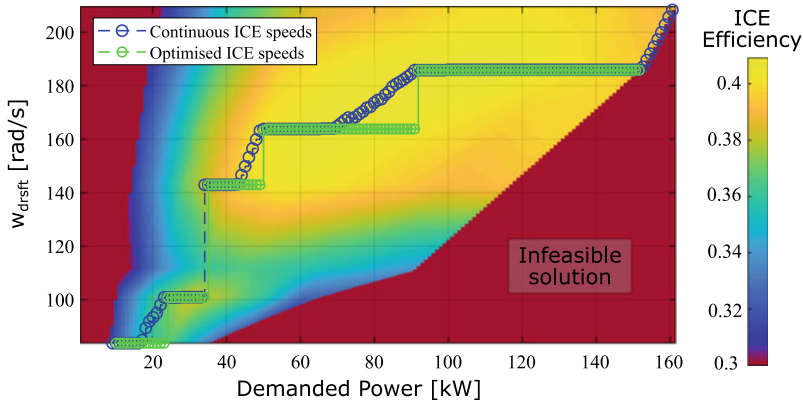
Fig. 14.11 GS optimal operation curve and efficiency map

on the power demanded to the GS) the instantaneous targets for the ICE rotational speed and corresponding mechanical torque [30].

14.4.1.7 Auxiliary Loads

The auxiliary loads, represented by the air conditioning, lights, and driving motors, is represented as an average of 12.5 kW for the optimization process. For the real fleet simulation, a random disruption of the auxiliary consumption has been implemented, as shown in Fig. 14.20.

The auxiliary loads in the case of the parallel configuration are powered by the BT. However, in the case of the series configuration, the auxiliary loads can be powered by the BT and the GS.

14.4.1.8 Energy Storage System Model

As it has been aforementioned in Section 14.3, the HEB fleet is composed of buses with BT pack and UC pack. Therefore, a model for each ESS has been developed as shown in Fig. 14.12 [30], the steady-state ESS equivalents [30].

The BT cell is represented by an ideal open-circuit voltage source [V_{OCBT} (V)] in series with the internal resistance [R_{BTcell} (Ω)]. For the SC model, a similar approach has been developed. In this case, an equivalent circuit of a capacitor with a capacitance C_{SCcell} (F) in series with a resistance $R_{int_S Ccell}$ (Ω) has been considered. For both models, it has been assumed that a string contains n_{BT} or n_{SC} BT or SC cells in series and the SC or BT pack groups m_{BT} and m_{SC} strings in parallel [30].

For the state of charge (SOC) estimation of the ESS ($SOC_{ESS}(k)$), for both BT and UC, coulomb counting method has been used [49]. In this modeling, the ESS current [$I_{ESS}(k)$ (A)] is calculated at each sampling (k), as follows:
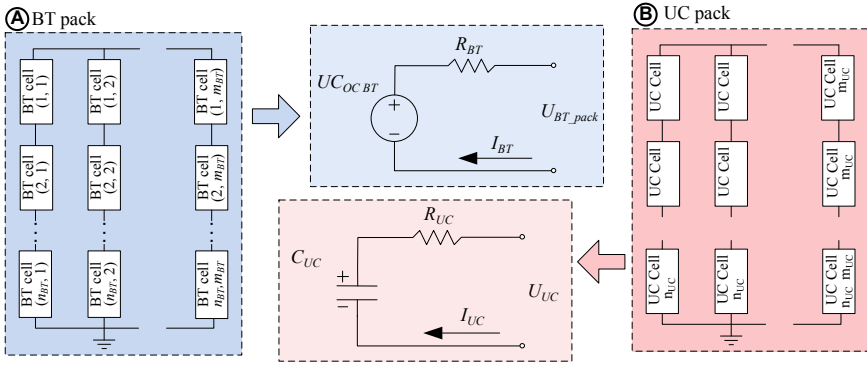
Fig. 14.12 **a** Battery pack and **b** UC pack electric models

$$I_{ESS}(k) = \frac{U_{ESS}(SOC_{ESS}(k))}{2 \cdot R_{ESS}} - \frac{\sqrt{U_{ESS}(SOC_{ESS}(k)) - 4 \cdot R_{ESS} \cdot P_{ESS}(k)}}{2 \cdot R_{ESS}} \quad \text{(A)}$$

(14.23)

where $U_{ESS}(k)$ (V) is the equivalent open-circuit voltage and $R_{ESS}(k)$ (Ω) is the equivalent internal resistance of the ESS, at pack level. The ESS model input is the power target $P_{ESS}(k)$ (W) generated by the dynamic programming (DP) function.

The SOC is updated at each sample as follows:

$$SOC_{ESS}(k+1) = SOC_{ESS}(k) - \frac{I_{ESS}(k+1)}{C_{ESS} \cdot 3600} \cdot 100 \quad (\%)$$

(14.24)

where C_{ESS} (Ah) is the ESS nominal capacity.

14.4.2 Optimization by Dynamic Programming

DP approach has been used to determine their optimal operation and performance for the fuel consumption minimization, based on the function developed in [50]. This approach has been commonly used as a baseline for benchmarking the proposed new EMSs for hybrid electric vehicles and for offline optimization [33, 51]. The optimization problem is based on the following cost function (J):

$$J = \sum_{k=0}^{N-1} \triangle m f_{ICE}(U(k)) \cdot T_s$$

(14.25)

where $\triangle m_f \cdot T_s$ is the fuel mass consumption at each time step ($T_s = 1$ s), determined by the torque (parallel configuration) or power (series configuration) split factor U, within the urban route length (N).

Table 14.3 Economic parameters

	Acronym	Value	Unit
Fuel cost	C_{fuel}	1.1	€/l
Cold start	k_{cs}	1.15	–
Energy electricity cost	C_{kWh}	0.088	€/kWh
Power electricity cost	C_{kW}	25.9	€/kW/year
Battery cost	C_{BT}	800	€/kWh
Ultracapacitor cost	C_{UC}	4000	€/kWh
ESS maintenance cost	$C_{ESS-Maint}$	500	€/year
Interest rate	I	2.5	%
Diesel density	ρ_{fuel}	0.832	kg/l
Calorific value	H_l	43.5e3	kJ/kg

14.4.3   Economic Model

An economic model has been developed, with the aim of evaluating the fleet life-time operation costs of the different configurations. For this evaluation fuel cost, recharging cost and BT replacement cost have been taken into account.

In Table 14.3, the economic section values are shown. These values have been considered for the aforementioned calculations [30, 52, 53].

14.4.3.1   Fuel Cost Calculation

The daily fuel consumption cost $\left[Fuel_{cost} \left(\frac{€}{day} \right) \right]$ is calculated by integrating the results obtained from the interpolation of Fig. 14.10 fuel consumption map as follows:

$$Fuel_{cost} = \sum_{k=1}^{p} \frac{k_{cs} \cdot mf_{ICE}(k) \cdot C_{fuel}}{\rho_{fuel}} \quad \left(\frac{€}{day} \right) \tag{14.26}$$

where k_{cs} (–) is the cold start $mf_{ICE}(k)$ $\left(\frac{kg}{s} \right)$ is the fuel mass flow and ρ_{fuel} $\left(\frac{kg}{m^3} \right)$ the density of the diesel.

14.4.3.2   Grid Cost Calculation

The grid costs are calculated according to the recharged energy in each round trip. The recharged cost $\left[Charging_{cost} \left(\frac{€}{day} \right) \right]$ is calculated as follows:

$$Charging_{cost} = \frac{C_{kW} \cdot P_{cha}}{365} + \sum_{k=1}^{p} \frac{P_{cha}(k) \cdot C_{kWh}}{3600} \quad \left(\frac{€}{day} \right) \tag{14.27}$$

where $C_{kW}\left(\frac{€}{kW}/year\right)$ is the cost of the power, $P_{cha}(k)$ (kW) is the power of the charger, $P_{cha}(k)$ (kW) is the power absorbed from the grid at each k discrete step, and $C_{kWh}\left(\frac{€}{kWh}\right)$ the referential energy cost of the grid.

14.4.3.3 Operation Costs of the Energy Storage System

In this subsection, the BT lifetime and BT operation cost models are presented. For a better understanding, it has been divided into two points, BT aging estimation and BT aging operation costs.

Battery Aging Estimation

In this subsection, the BT lifetime (γ) and the number of replacement calculation method is described. For this calculation, BT calendar degradation, maximum full equivalent cycles (FECs), and BT cycling degradation methods have been taken into account.

$$\gamma = min[\gamma_{cal}, FEC, \gamma_{cyc}] \tag{14.28}$$

where γ_{cal} is the number of years by means of the calendar degradation, FEC is the accounted FEC of the BT, and γ_{cyc} is the degradation by means of the BT operation. The calendar degradation and FEC are fixed values provided by the BT manufacturer. However, the BT cycling degradation has to be evaluated. In the following lines, the BT cycling degradation calculation is introduced.

BT cycling degradation is calculated based on a rainflow cycle counting algorithm [54] and Wöhler curve-based method [30]. The Wöhler curve-based method is a fatigue analysis, commonly used for BT aging estimations [53, 55, 56].

The Wöhler method lies on the number of NE_{ievt} events—in this case depth of discharge (DOD)—that can occur until the BT reaches its end-of-life (EOL).

The lifetime lost (LL_{ievt}) calculation is done by the relation of the accounted (NE_{ievt}) and the maximum number of events (NE_{ievt}^{max}) that the BT can withstand, expressed as follows:

$$LL_{ievt} = \frac{NE_{ievt}}{NE_{ievt}^{max}} \tag{14.29}$$

The NE_{ievt} are accounted by means of the rainflow algorithm (Fig. 14.13), with steps of 1% of DOD.

The NE_{ievt}^{max} are extracted from the Wöhler curve. Figure 14.14 [30] depicts an example of a Wöhler curve. The data for building this curve can be obtained directly from the manufacturers or by means of laboratory tests.
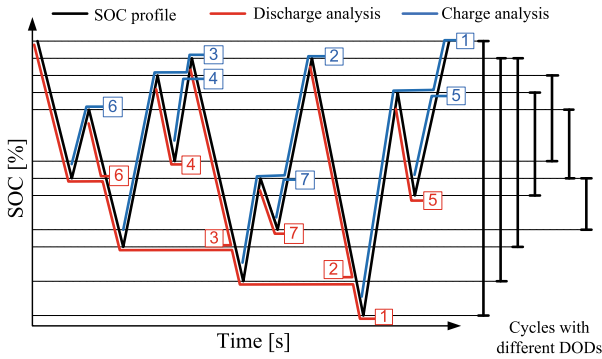
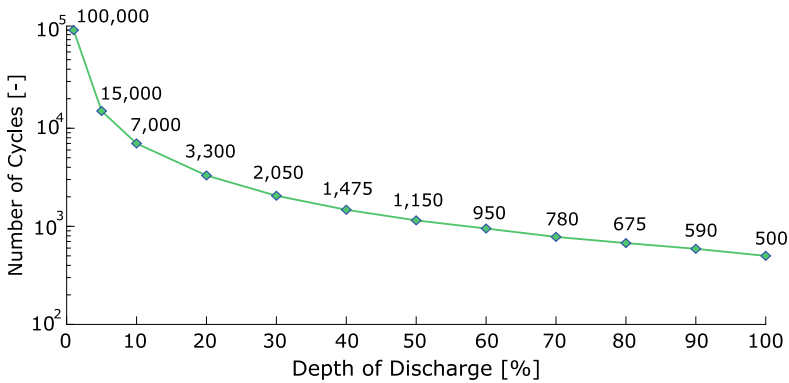**Fig. 14.13** Rainflow charging/discharging cycle counting algorithm

Fig. 14.14 Wöhler curve example

For determining the total lifetime loss (*LL*) in the whole range of DODs (from 0 to 100%), the sum of all the events in the cycling evaluated period has to be calculated as follows [30]:

$$LL = \sum_{ievt} LL_{ievt} \qquad (14.30)$$

Finally, considering the evaluated SOC profile's time period can be calculated as the inversion of LL, the total cycling lifetime (γ_{cyc}) can be calculated, typically defined in years:

$$\gamma_{cyc} = \frac{1}{\sum_{ievt=1}^{100} \left(\frac{NE_{ievt}}{NE_{ievt}^{max}} \right)} \qquad (14.31)$$

Battery Aging Operations Costs

Therefore, based on the previously estimated BT aging, the BT cost (BT_{cost}) is calculated as follows:

$$BT_{cost} = \frac{BT_{Maint} + BT_{Ini} + BT_{Rep}}{365} \quad \left(\frac{\euro}{\text{day}}\right) \tag{14.32}$$

where BT_{Maint} is the annualized maintenance cost, BT_{Ini} is the annualized capital cost related to the initial investment of the BT pack, and BT_{Rep} is the annualized replacement cost of the BT pack. These costs are further explained in the following lines.

$$BT_{Ini} = (C_{kWh-BT/UC} \cdot E_{ESS}) \cdot CRF \quad \left(\frac{\euro}{\text{year}}\right) \tag{14.33}$$

where $C_{kWh-BT/UC}$ ($\frac{\euro}{\text{kWh}}$) is the referential cost of the BT or UC technology, E_{ESS} (kWh) is the capacity of the installed BT pack, and CRF is the capital recovery factor. The latter equation allows annualizing the cost considering the lifetime of the whole system:

$$CRF = \frac{I \cdot (I + T)^T}{1 + (I + T)^T} \quad \left(\frac{1}{\text{year}}\right) \tag{14.34}$$

where I (%) and T (years) refer to the interest rate and the lifetime of the whole system, respectively.

The expression of BT_{Rep} is calculated as follows:

$$BT_{Rep} = \sum_{k=1}^{n_{Rep}} \frac{C_{kWh-ESS} \cdot E_{ESS} \cdot CRF}{(I + 1)^{k \cdot \gamma}} \quad \left(\frac{\euro}{\text{year}}\right) \tag{14.35}$$

where n_{Rep} is the number of ESS replacements calculated based on the previously calculated γ and years of operation.

14.5   Fleet Learning Methodology

In order to overcome the lack of an existing methodology for energetic behavior improvement at fleet level, the methodology shown in Fig. 14.17 has been proposed. The methodology is composed of six different stages, which are explained in the following subsections.

14.5.1 Stage 1: *Design Stages*

The first stage lies for the design stage. This stage is composed of three substages, which are used to create the urban profile, optimize the operation, and generate the EMS.

Stage 1.1: *Urban Route Profiles*
In this substage, the urban routes have been generated from a database of 112 standardized profiles, randomly selected to complete a daily route. In order to enlarge this database, a mixture of profiles has been done, simulating a round trip, as depicted in Fig. 14.15.

Stage 1.2: *Optimal Operation*
For simulating the optimal operation, DP approach has been used (see Section 14.4.2). The obtained (power or torque) split factor, SOC profile, and fuel consumption have been used as a baseline for the analysis and comparison of the whole fleet. In addition, lookup tables (LUT) have been generated and used to provide optimal operation target, as shown in Fig. 14.16.

Stage 1.3: *Energy Management Strategy*
The developed EMS for each HEB has been a rule-based (RB) strategy. For evaluating the efficiency increase of DP, first an RB strategy without the LUT has been used, for having all the bus with the same EMS. After this evaluation, the following personalized EMS for each bus has been used (Figure 14.17).
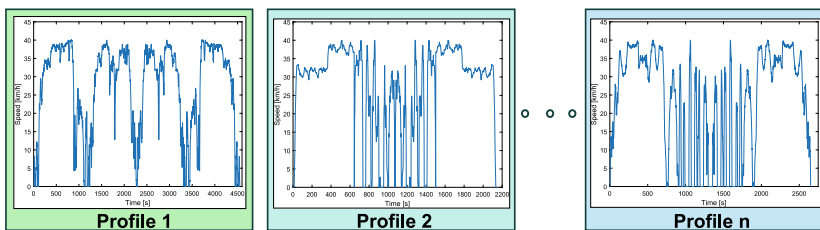
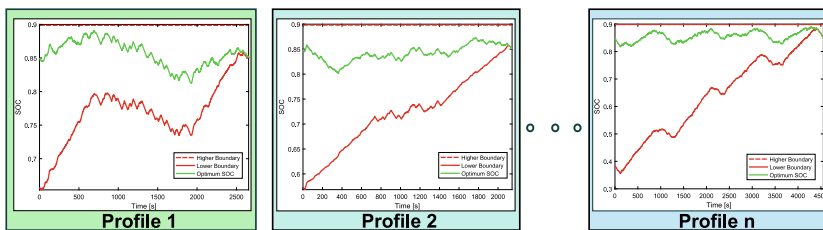**Fig. 14.15** Random new "*n*" driving profiles

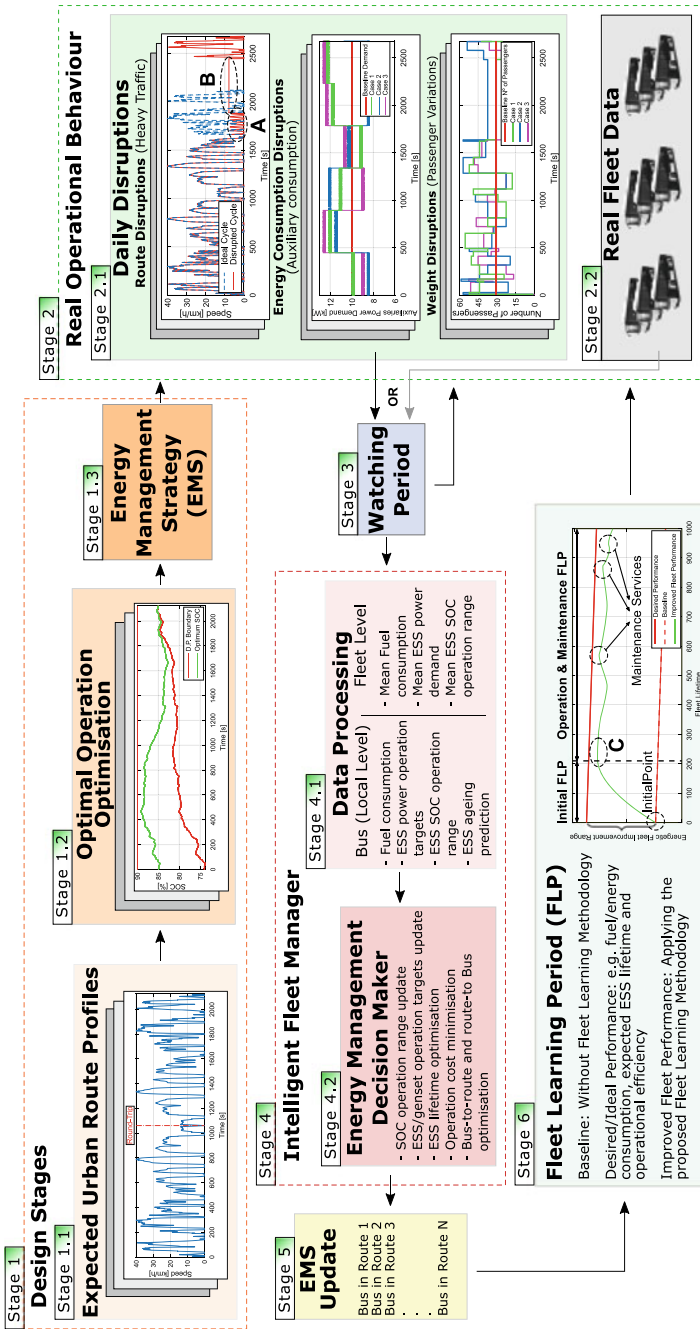**Fig. 14.16** Optimized random "*n*" driving profiles

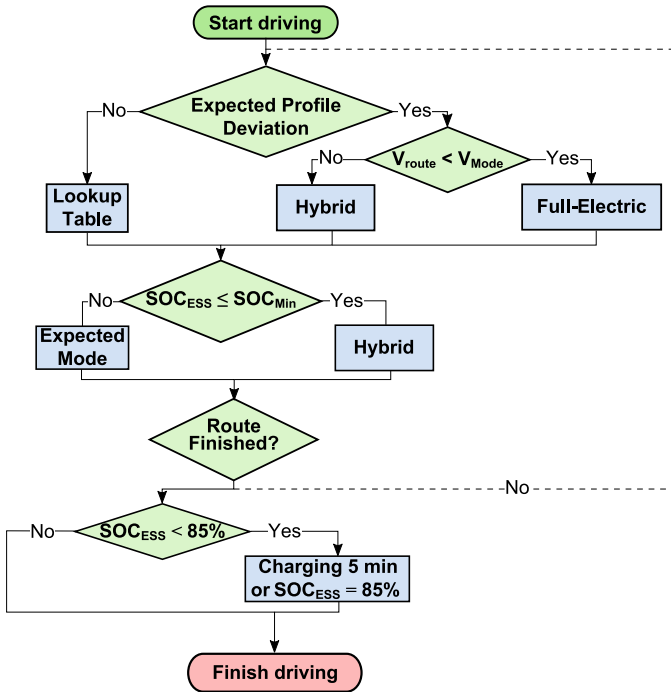**Fig. 14.17** Proposed fleet learning methodology

Fig. 14.18 Rule-based strategy

This RB strategy makes decisions according to the deviation from the expected profile (tolerance of 2%), as shown in Fig. 14.18. When the "real" profile is within the limits, LUTs are applied (driven by the distance).

When the "real" profile is outside the expected route, the following decisions are taken. The first decision is taken according to the speed of the bus (being V_{Mode} 10 km/h), using full-electric mode—all the energy is provided by the ESS—when the speed is lower than V_{Mode} and hybrid mode—providing the required energy and recharging the ESS from the GS—when the speed is higher than V_{Mode}.

The following rule is above the aforementioned ones. It decides based on the current SOC. If the current SOC is below a defined minimum SOC (being the threshold of the SOC 60% in the case of the BT pack and 40% in the case of UCs), the driving mode is modified to the hybrid. If it is above the threshold, nothing is done.

The last decision is taken based on the SOC when the route is finished. When this final SOC is below 85%, the HEB is connected to the grid for 5 min in order to recharge the ESS. The charger used for the BTs is a 30 kW charger and for the UCs a 60 kW one [57].

14.5.2   Stage 2: *Real Operational Behavior*

These stages are used to simulate the real operation behavior or to extract the data from a real fleet.

Stage 2.1: *Daily Disruptions*
A daily trip journey is composed of around 16 h. For simulating "real" trips, some disruptions have been randomly introduced to these routes. For this scenario, three disruptions have been considered.

- Road disruptions: Heavy traffic has been simulated increasing the number of unscheduled stops and delays in the daily route for evaluating the energetic behavior. In Fig. 14.19, **A** reflects the simulation of a traffic jam and **B** the heavy traffic after a traffic jam.
- Power disruptions: These disruptions have been simulated increasing/decreasing the auxiliary power consumption, related to temperature variations, as shown in Fig. 14.20.
- Weight disruptions: These disruptions have been represented with the passenger movement on each bus stop, as shown in Fig. 14.21.

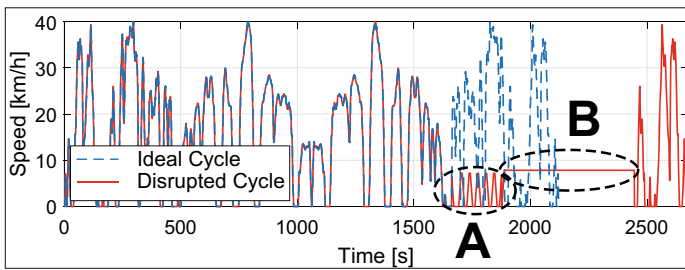These disrupted routes have been simulated applying the aforementioned RB strategy.
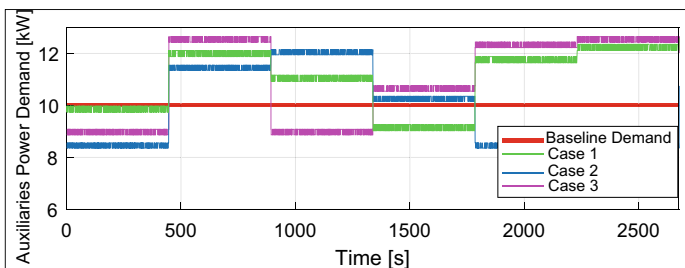

Fig. 14.19 Traffic jam disruption

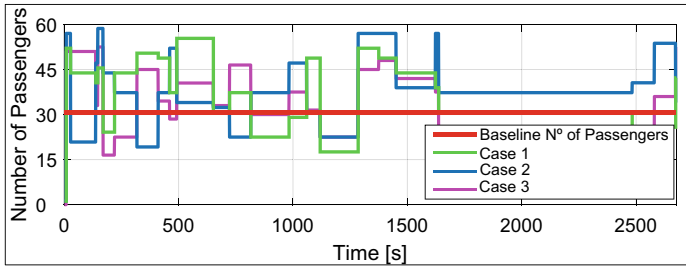

Fig. 14.20 Auxiliary consumption disruption

Fig. 14.21 Passenger variation weight disruption

Stage 2.2: *Real Fleet Data*
This stage depicts the possibility of extracting the data directly from a real fleet, instead of simulating it, by means of stage 2.1.

14.5.3   Stage 3: *Watching Period*

In this stage, a time period has to be set, with the aim of collecting enough data for the processing stage. This period can be set from weeks until years, depending on the data analysis type to be focus on. Based on the period time, two watching periods are differentiated the short- and long-term watching periods.

On the one hand, the short-term watching period can be set from weeks to months and it is focused on the operation performance improvement. On the other hand, the long-term period is set from several months to years. This watching period is set with the aim of evaluating components that need longer evaluation periods for making a decision, such as the BT lifetime extension or the re-engineering of the power train.

14.5.4   Stage 4: *Fleet Learning Period*

In these stages, the collected data from the fleet is processed and analyzed, for subsequently utilizing as a basis for making energy management decisions.

Stage 4.1: *Data Processing*
The collected data in the watching period stage has been processed locally (at bus level) and at fleet level at this stage. This processed data gives the information to compare fuel consumption, ESS power operation targets, and ESS SOC operation range. On the one hand, the processed data at local level gives the information on each bus. On the other hand, at fleet level, the data is processed calculating the overall mean values. At the local level in each HEB, an aging model can be implemented to the ESS for predicting the aging.

Stage 4.2: *Energy management decision-maker*

Taken those buses with the best energetic behavior, this decision-maker stage decides the changes to be applied.

- *SOC operation range update*: Analyzing the ESS SOC operation range allows to adapt SOC range to each urban route demand. This helps to optimize the BT operation.
- *ESS/GS power operation targets update*: According to each urban route power demand, these targets are updated, in order to optimize the EMS.
- *ESS lifetime maximization*: Based on the result of the ESS aging prediction, the BT operation is updated, to prolong the ESS lifetime.
- *Operation cost minimization*: Possible resizing of the ESS, ICE, or EM.
- *Bus-to-route optimization*: At first step, adapting the bus to the route with the optimized LUTs.
- *Route-to-bus optimization*: After a time period, adapting the route to each bus, switching the buses having a worse SOH and more intensive routes, with those buses having a better SOH and less-intensive routes.

14.5.5   Stage 5: *EMS Update*

Based on the taken decisions in the previous 4.2 stage, the EMS is updated for each individual bus.

14.5.6   Stage 6: *Fleet Learning Period*

This feedback stage is used to evaluate the whole fleet efficiency every time the EMS has been updated, in order to analyze the impact of the taken decision. A learning period is determined by this feedback process, obtaining the fleet learning curve, as shown in Fig. 14.22.

As depicted in Fig. 14.22A, a baseline is set, which will be the prediction of the fleet energetic efficiency evolution if no measures are taken. For the baseline starting point, an initial energetic efficiency point has to be set, as the fleet is composed of HEBs with different aging times. A desired performance target line is set, which will represent the maximum energetic fleet improvement range.

Once the boundaries of the fleet learning period are set, based on the historical data the improved fleet performance is represented. Analyzing Fig. 14.22B, it is noteworthy that the fleet learning curve is divided in two periods: the initial fleet learning period and the operation and maintenance fleet learning period.

The initial fleet learning period is split into two learning approaches: the self-learning and the fleet learning approach. The self-learning approach is focused at single vehicle level, optimizing the vehicle based on the route. On the contrary,
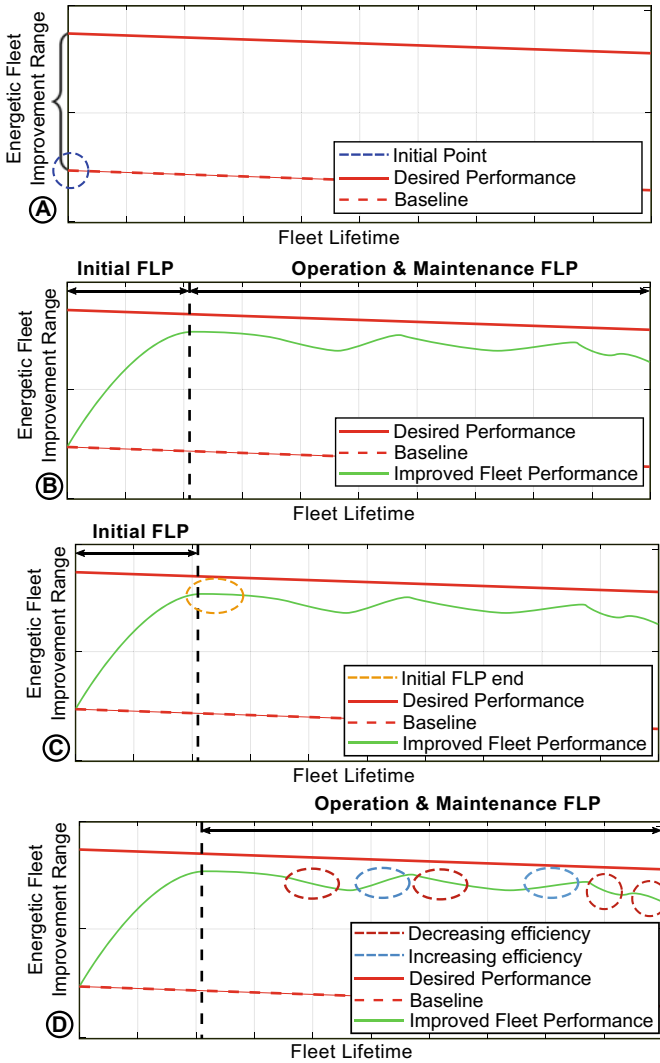
Fig. 14.22 Standardized cycles combination and round-trip generation

the fleet learning approach is based on the whole fleet information, learning from those with the best energetic behavior. This learning period can be concluded in two situations: First, when the fleet overall efficiency has reached (or approximated to) the desired efficiency target and it has been stabilized; and second, when the curve has changed the tendency, as shown in Fig. 14.22C.

The final learning period is based on the operation and maintenance period. This stage starts when there is a decreasing tendency. The decreasing tendency of the overall efficiency is due to factors such as ESS SOH decrease. In this new stage, some

increasing tendencies can be observed, as depicted in Fig. 14.22D, which represent the efficiency improvement after operation and maintenance modifications, such as ESS replacement or auxiliary equipment efficiency increase.

14.6   Results and Analysis

In order to validate the proposed "fleet learning" methodology, a simulation of the fleet described in Table 14.1 has been carried out. The watching period has been defined of a week (7 days), with around 16 hours a daily operation. The DP approach results (obtained from the urban routes without perturbations) have been used as baseline value for contrasting the disrupted results. The new proposed approach has been applied on the disrupted urban routes.

Figure 14.23 depicts the mean fuel consumption of both approaches. In the three configurations, the proposed approach shows a lower fuel consumption compared to the baseline. This reduction is given as a result of a constraint only applied to the DP approach. This constraint ensures to start and finish at the same SOC level (85%). Consequently, there is a more intensive use of the ICE and the GS in the parallel and series configurations, respectively.

Figures 14.24 and 14.25 show the results regarding the ESS information (ESS energy consumption and the DOD respectively). In the proposed approach, in contrast to a less fuel consumption, it is noteworthy a more intensive use of the ESS. In the HEB with a BT pack, the ESS use is around the double from the baseline value (Par 180% and Ser1 271%). In the case of the HEB with UCs pack, the DOD has a minor increase. This is due to the fact that UCs have higher c-rates. As a result of this characteristic, the charging and discharging process is faster. In addition, as UCs have less capacity besides BTs, their use is more intensive.

In the proposed approach, when the final SOC is below 85%, for reaching the energetic balance, a charging process is carried out (see Fig. 14.18). The recharged energy in each power train configuration is shown in Fig. 14.26. An additional
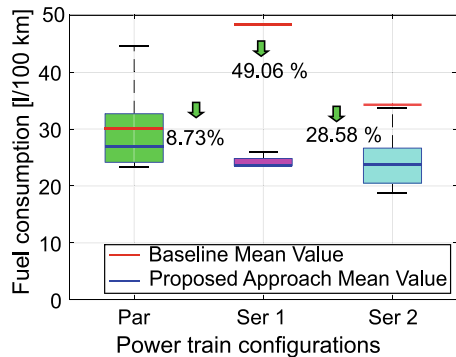
Fig. 14.23 Fleet fuel consumption
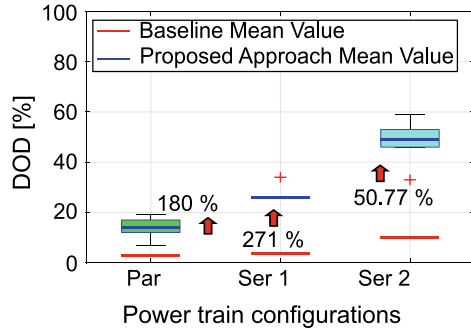
Fig. 14.24 Fleet ESS analyzed DODs

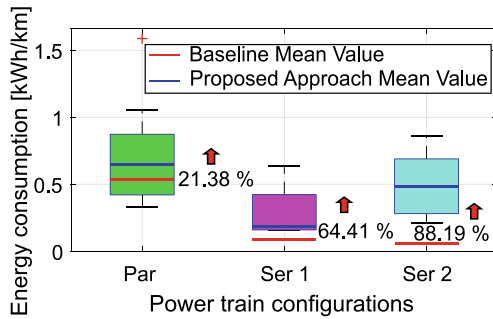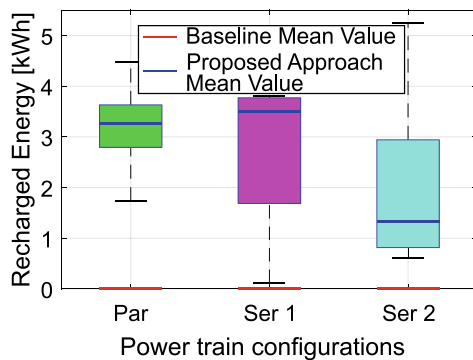**Fig. 14.25** Fleet ESS energy consumption

Fig. 14.26 Fleet recharged energy from the charger

flexibility degree is added with the charging process, consequently obtaining a more intensive ESS use in the proposed approach than in the DP approach.

Comparing the three different power train configurations, it is important to highlight that the parallel configuration is less flexible than the series one. This is due to the fact that both the EM and ICE are fixed to the same axis. This reduces the flexibility degree of recharging the ESS, having the chance of recharging with regenerative braking, coasting operation mode, and from the grid when the route is finished. On the contrary, the series configuration has the possibility of recharging the BT at any

Fig. 14.27 Fleet mean
operation costs per day
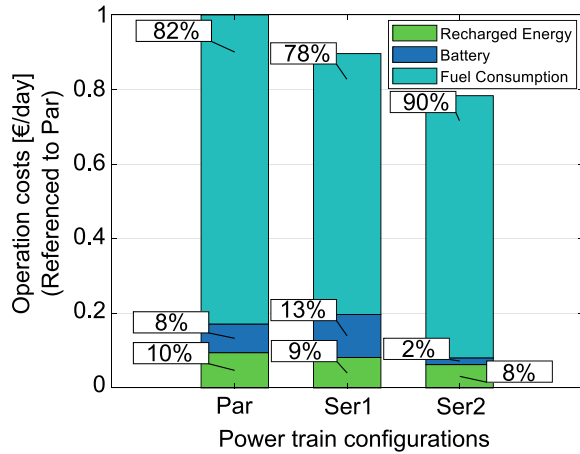
given time, while driving with the GS, with the regenerative braking and from the
grid when the route is finished.

In Fig. 14.27, the operation costs per day are shown. For this calculation, the
fuel consumption, BT replacement rate (lifetime estimation obtained with a Wöhler
degradation estimation technique [49]), and charging costs, as explained in Sub-
section 14.4.3, have been considered. It is noteworthy the fuel consumption high
influence on the operation costs, reaching the value up to 90% of the total daily
operation costs.

In this regard, the fuel consumption, BT replacement rate, and charging costs
throughout the buses lifetime have been calculated as shown in Fig. 14.28. As it has
been aforementioned, it is to highlight the fuel consumption high influence on the
operation costs. Consequently, despite having the highest replacement rate of BT
packs on the series configuration, the fuel consumption reduction makes to have the
highest operation cost reduction.

In order to quantify the obtained improvement from the proposal, Fig. 14.29
depicts the examined fleet energetic efficiency. For the fleet learning improvement
quantification, results from the operation costs have been analyzed. This fleet learn-
ing period has been evaluated from the historical data in two periods, T1 and T2.
T1 represents the obtained improvement with the DP optimization compared to the
aforementioned RB strategy without the LUT, with an increase up to 46%, simi-
lar to other works such as [58]. The second evaluation period, T2, is the energetic
fleet improvement applying the proposed methodology and obtaining an increase of
18.7%. This last improvement mainly is due to the decrease of the fuel utilization, as
depicted in Fig. 14.23. The first optimization has a greater improvement besides the
proposed approach, due to the low adaptability of the EMS without the LUT. Finally,
an estimation of the upcoming learning period is depicted. The downward and upward
efficiency tendencies are due to ESS SOH decrease or overall bus efficiency decrease
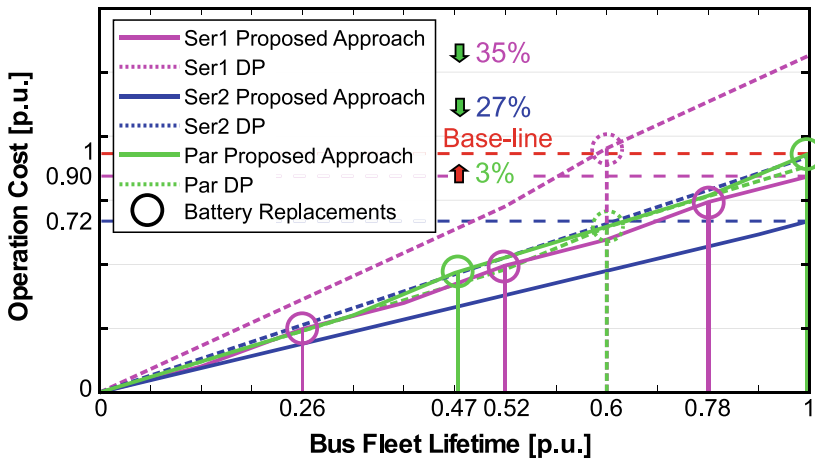and efficiency increase after maintenances, respectively.

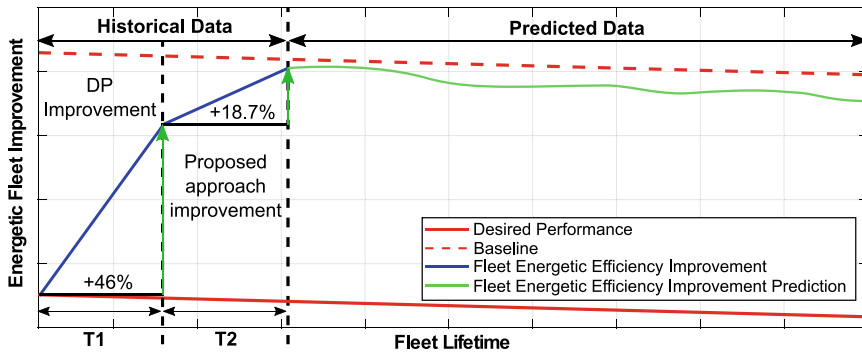**Fig. 14.28** Buses fleet lifetime operation cost estimation

Fig. 14.29 Fleet learning period for the fleet

14.7   Conclusions

In this chapter, a state-of-the-art of the different proposed fleet management approaches in the literature was presented. An overview of the HEBs electrical model, DP optimization, and the used economical model for the fleet operation costs calculation was described. In order to fill the gap of the no existing methodology for the overall fleet energetic efficiency improvement, a "fleet learning" methodology was proposed. This methodology approaches the design and operation stages; in this, last is taken into account the whole fleet improvement, obtaining an energetic improvement on the fleet up to 18.7%. In order to simulate the real operation, daily disruptions (traffic jams, auxiliary consumption, and passenger variations) have been simulated.

The fuel consumption decrease in the proposed approach is up to 49.06%, consequently having a more intensive use of the ESS. The ESS consumption increases up to 88%, underlining the DOD increase in the three configurations (increasing up to 271%). It is noteworthy the influence of the power train configuration and the requirement of the urban route, at fleet operation level, highlighting the flexibility of the series HEB and its operation optimization, due to the possibility to operate the GS in the optimal speed (ICE) and torque (EG).

In the series configurations, the lifetime operation costs are reduced, not obtaining the same result in the parallel configuration, as it has the least fuel consumption reduction. The highest cost reduction is given in the power train with the highest fuel consumption reduction, the series configuration with BT (decreasing the fuel consumption up to 49.06%). Despite this configuration has the highest BT replacement rate, due to a deeper ESS DOD, the fuel consumption has a greater influence on the operation costs.

On the ongoing research, an intelligent decision-maker (based on machine learning techniques, stage 4.2 in Fig. 14.17) will be developed.

References

1. Key world, energy trends, except from: world energy balances. Technical report, International Energy Agency (2016)
2. Lopez, J.A., Herrera, V. I., Camblong, H., Milo, A., Gaztanaga, H.: Energy management improvement based on fleet learning for hybrid electric buses. In: 2018 IEEE Vehicle Power and Propulsion Conference (VPPC), Chicago, IL, pp. 1–6 (2018). https://doi.org/10.1109/VPPC.2018.8605025
3. Heavy-duty vehicles CO_2 emissions and fuel efficiency. http://www.europarl.europa.eu/legislative-train/theme-resilient-energy-union-with-a-climate-change-policy/file-heavy-duty-vehicles-co2-emissions-and-fuel-efficiency, 20-04-2018. Accessed 21 May 2018
4. Vedlugaite, D.: Greenhouse gas emissions from transport. Technical report, European Environment Agency (2018)
5. Bulc, V.: An overview of electric buses in Europe. Technical report (2017)
6. Reducing CO_2 emissions from heavy-duty vehicles. https://ec.europa.eu/clima/policies/transport/vehicles/heavy_en. Accessed 22 Aug 2018
7. Steriu, M.: Local public transport in the European Union, UITP. Technical report, Advancing Public Transport (2016)
8. Tzeng, G.H., Lin, C.W., Opricovic, S.: Multi-criteria analysis of alternative-fuel buses for public transportation. Energy Policy **33**(11), 1373–1383 (2005)
9. Nurhadi, L., Borén, S., Ny, H.: A sensitivity analysis of total cost of ownership for electric public bus transport systems in Swedish medium sized cities. Transp. Res. Procedia **3**(July), 818–827 (2014) [Online]. http://dx.doi.org/10.1016/j.trpro.2014.10.058
10. Glotz-Richter, M., Koch, H.: Electrification of public transport in cities (Horizon 2020 ELIPTIC Project). Transp. Res. Procedia **14**, 2614–2619 (2016) [Online]. http://dx.doi.org/10.1016/j.trpro.2016.05.416
11. Sen, B., Ercan, T., Tatari, O.: Does a battery-electric truck make a difference? Life cycle emissions, costs, and externality analysis of alternative fuel-powered Class 8 heavy-duty trucks in the United States. J. Clean. Prod. **141**, 110–121 (2017) [Online]. http://dx.doi.org/10.1016/j.jclepro.2016.09.046

12. Tsita, K.G., Pilavachi, P.A.: Decarbonizing the Greek road transport sector using alternative technologies and fuels. Therm. Sci. Eng. Prog. **1**, 15–24 (2017) [Online]. http://dx.doi.org/10.1016/j.tsep.2017.02.003

13. Bakker, S., Konings, R.: The transition to zero-emission buses in public transport—the need for institutional innovation. Transp. Res. Part D: Transp. Environ. (March), 0–1 (2017) [Online]. http://dx.doi.org/10.1016/j.trd.2017.08.023

14. Çabukoglu, E., Georges, G., Küng, L., Pareschi, G., Boulouchos, K.: Battery electric propulsion: an option for heavy-duty vehicles? Results from a Swiss case-study. Transp. Res. Part C: Emerg. Technol. **88**(September 2017), 107–123 (2018) [Online]. https://doi.org/10.1016/j.trc.2018.01.013

15. Mareev, I., Becker, J., Sauer, D.U.: Battery dimensioning and life cycle costs analysis for a heavy-duty truck considering the requirements of long-haul transportation. Energies **11**(1) (2018)

16. Lajunen, A.: Energy consumption and cost-benefit analysis of hybrid and electric city buses. Transp. Res. Part C: Emerg. Technol. **38**, 1–15, 2014. [Online]. http://dx.doi.org/10.1016/j.trc.2013.10.008

17. Ribau, J.P., Silva, C.M., Sousa, J.M.: Efficiency, cost and life cycle CO_2 optimization of fuel cell hybrid and plug-in hybrid urban buses. Appl. Energy **129**, 320–335 (2014)

18. Mahmoud, M., Garnett, R., Ferguson, M., Kanaroglou, P.: Electric buses: a review of alternative powertrains. Renew. Sustain. Energy Rev. **62**, 673–684 (2016)

19. Kunith, A., Mendelevitch, R., Goehlich, D.: Electrification of a city bus network: an optimization model for cost-effective placing of charging infrastructure and battery sizing of fast charging electric bus systems. SSRN Electron. J. (2016) [Online]. https://www.ssrn.com/abstract=2782417

20. Randall, T.: Here's how electric cars will cause the next oil crisis. https://www.bloomberg.com/features/2016-ev-oil-crisis/, 05-12-2016. Accessed 26 April 2018

21. McKinsey & Company: Urban buses: alternative powertrains for Europe, pp. 1–57 (2012)

22. Pihlatie, M., Kukkonen, S., Halmeaho, T., Karvonen, V., Nylund, N.O.: Fully electric city buses—the viable option. In: IEEE International Electric Vehicle Conference, IEVC 2014 (2015)

23. Palmer, K., Tate, J.E., Wadud, Z., Nellthorp, J.: Total cost of ownership and market share for hybrid and electric vehicles in the UK, US and Japan. Appl. Energy **209**(July 2017), 108–119 (2018) [Online]. https://doi.org/10.1016/j.apenergy.2017.10.089

24. Ranta, M., Pihlatie, M., Pellikka, A., Laurikko, J., Rahkola, P., Anttila, J.: Analysis and comparison of energy efficiency of commercially available battery electric buses (2017)

25. Manzie, C., Watson, H., Halgamuge, S.: Fuel economy improvements for urban driving: hybrid vs. intelligent vehicles. Transp. Res. Part C: Emerg. Technol. **15**(1), 1–16 (2007)

26. Mesgarpour, M., Landa-Silva, D., Dickinson, I.: Overview of telematics-based prognostics and health management systems for commercial vehicles. Commun. Comput. Inf. Sci. **395**, 123–130 (2013)

27. Muthuramalingam, S., Bharathi, A., Rakesh Kumar, S., Gayathri, N., Sathiyaraj, R., Balamurugan, B.: IoT based intelligent transportation system (IoT-ITS) for global perspective: a case study. In: Internet of Things and Big Data Analytics for Smart Generation. Intelligent Systems Reference Library, ch. 13, pp. 279–300. Springer, Cham (2019)

28. Mehar, S., Zeadally, S., Remy, G., Senouci, S.M.: Sustainable transportation management system for a fleet of electric vehicles. IEEE Trans. Intell. Transp. Syst. **16**(3), 1401–1414 (2015) [Online]. http://ieeexplore.ieee.org/document/6975190/

29. Rosario, L., Luk, P.C.K.: Applying management methodology to electric vehicles with multiple energy storage systems. In: Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, ICMLC 2007, vol. 7, no. August, pp. 4223–4230 (2007)

30. Herrera, V.: Optimized energy management strategies and sizing of hybrid storage systems for transport applications (2017)

31. Zhang, P., Yan, F., Du, C.: A comprehensive analysis of energy management strategies for hybrid electric vehicles based on bibliometrics. Renew. Sustain. Energy Rev. **48**(205), 88–104 (2015) [Online]. http://dx.doi.org/10.1016/j.rser.2015.03.093

32. Sabri, M.F.M., Danapalasingam, K.A., Rahmat, M.F.: A review on hybrid electric vehicles architecture and energy management strategies. Renew. Sustain. Energy Rev. **53**, 1433–1442 (2016) [Online]. http://dx.doi.org/10.1016/j.rser.2015.09.036

33. Marina Martinez, C., Hu, X., Cao, D., Velenis, E., Gao, B., Wellers, M.: Energy management in plug-in hybrid electric vehicles: recent progress and a connected vehicles perspective. IEEE Trans. Veh. Technol. **99**, 1–1 (2016) [Online]. http://ieeexplore.ieee.org/document/7496906/

34. Thong, S.T.S., Chua, T.H., Rahman, T.A.: Intelligent fleet management system with concurrent GPS & GSM real-time positioning technology. In: ITST 2007—7th International Conference on Intelligent Transport Systems Telecommunications, Proceedings, pp. 136–141 (2007)

35. Balaji, P.G., Sachdeva, G., Tham, C.K.: Multi-agent system based urban traffic management. In: 2007 IEEE Congress on Evolutionary Computation (CEC 2007), vol. 117576, pp. 1740–1747 (2007)

36. HomChaudhuri, B., Lin, R., Pisu, P.: Hierarchical control strategies for energy management of connected hybrid electric vehicles in urban roads. Transp. Res. Part C: Emerg. Technol. **62**, 70–86 (2016) [Online]. http://dx.doi.org/10.1016/j.trc.2015.11.013

37. Grantner, J., Bazuin, B., Dong, L., Al-shawawreh, J., Hathaway, R., Fajardo, C., Castanier, M.P., Hussain, S.: Condition based maintenance for light trucks. In: Conference Proceedings—IEEE International Conference on Systems, Man and Cybernetics, pp. 336–342 (2010)

38. Haycock, R., Ferguson, C.: Creating demand in fleets for EVs, infrastructure and demand driven energy supply through enabling software that can significantly improve the TCO model of the complete system. In: 2013 World Electric Vehicle Symposium and Exhibition, EVS 2014, pp. 1–9 (2014)

39. Barré, A., Suard, F., Gerard, M., Riu, D.: Battery capacity estimation and health management of an electric vehicle fleet. In: 2014 IEEE Vehicle Power and Propulsion Conference (VPPC), vol. 16, no. 4, pp. 1–6. IEEE (2014) [Online]. http://ieeexplore.ieee.org/document/7007025/

40. Eisner, J., Funke, S., Storandt, S.: Optimal route planning for electric vehicles in large networks. In: AAAI, pp. 1108–1113 (2011) [Online]. http://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/viewPDFInterstitial/3637/4005

41. Sachenbacher, M., Leucker, M., Artmeier, A., Haselmayr, J.: Efficient energy-optimal routing for electric vehicles. In: Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, no. January 2011, pp. 1402–1407 (2011)

42. Remy, G., Mehar, S., Sophy, T., Senouci, S.M., Jan, F., Gourhant, Y.: Green fleet management architecture: application to economic itinerary planning. In: 2012 IEEE Globecom Workshops, GC Wkshps 2012, no. December, pp. 369–373 (2012)

43. Duchrow, T., Schroer, M., Griesbach, B., Kasperski, S., Bermpohl, F.M.G., Kramer, S., Kirchner, F.: Towards electric mobility data mining. In: 2012 IEEE International Electric Vehicle Conference, no. March, pp. 1–6 (2012) [Online]. http://ieeexplore.ieee.org/document/6183199/

44. Qin, H., Zhang, W.: Charging scheduling with minimal waiting in a network of electric vehicles and charging stations. In: Proceedings of the Eighth ACM International Workshop on Vehicular Inter-networking—VANET '11, no. May 2014, p. 51 (2011) [Online]. http://dl.acm.org/citation.cfm?doid=2030698.2030706

45. Rogge, M., van der Hurk, E., Larsen, A., Sauer, D.U.: Electric bus fleet size and mix problem with optimization of charging infrastructure. Appl. Energy **211**(Nov 2017), 282–295, 2018 [Online]. https://doi.org/10.1016/j.apenergy.2017.11.051

46. Wittmann, M., Lohrer, J., Betz, J., Jäger, B., Kugler, M., Klöppel, M., Waclaw, A., Hann, M., Lienkamp, M.: A holistic framework for acquisition, processing and evaluation of vehicle fleet test data. In: IEEE Conference on Intelligent Transportation Systems, pp. 1–7. Proceedings, ITSC (2017)

47. Guzzella, L., Sciarretta, A.: Vehicle Propulsion Systems (2005)

48. Ehsani, M., Gao, Y., Gay, S.E., Emadi, A.: Modern Electric, Hybrid Electric, and Fuel Cell Vehicles. CRC Press (2005)

49. Herrera, V.I., Milo, A., Gaztañaga, H., Etxeberria-Otadui, I., Villarreal, I., Camblong, H.: Adaptive energy management strategy and optimal sizing applied on a battery-supercapacitor based tramway. Appl. Energy **169**, 831–845 (2016)

50. Sundstr, O., Guzzella, L.: A generic dynamic programming Matlab function. In: 18th IEEE International Conference on Control Applications, Saint Petersburg, Russia, no. 7, pp. 1625–1630 (2009)

51. Silvas, E., Hofman, T., Murgovski, N., Etman, L.F.P., Steinbuch, M.: Review of optimization strategies for system-level design in hybrid electric vehicles. IEEE Trans. Veh. Technol. **66**(1), 57–70 (2017)

52. Lajunen, A.: Lifecycle costs and charging requirements of electric buses with different charging methods. J. Clean. Prod. **172**, 56–67 (2018) [Online]. https://doi.org/10.1016/j.jclepro.2017.10.066

53. Herrera, V., Milo, A., Gaztañaga, H., Camblong, H.: Multi-objective Optimization of Energy Management and Sizing for a Hybrid Bus with Dual Energy Storage System (2016)

54. Dufo, R.: Dimensionamiento y control optimo de sistemas híbridos aplicando algorítmos evolutivos (2007)

55. Facinelli, W.A.: Modeling and Simulation of Lead-Acid Batteries for Photovoltaic Systems (1983)

56. Sauer, D.U., Wenzl, H.: Comparison of different approaches for lifetime prediction of electrochemical systems-using lead-acid batteries as example. J. Power Sources **176**(2), 534–546 (2008)

57. Siemens: Charge your future—with the Siemens eBus charging infraestructure. https://w3.siemens.com/topics/global/de/elektromobilitaet/PublishingImages/ladetechnik-busse/pdf/ebus-brochure-en.pdf

58. Wang, R., Lukic, S.M.: Dynamic programming technique in hybrid electric vehicle optimization. In: IEEE International Electric Vehicle Conference, pp. 1–8 (2012)