Doubly Fed Induction Generators

Control for Wind Energy

Edgar N. Sanchez
Riemann Ruiz-Cruz

CRC Press
Taylor & Francis Group

Doubly Fed Induction Generators

CONTROL FOR WIND ENERGY

AUTOMATION AND CONTROL ENGINEERING
A Series of Reference Books and Textbooks

Series Editors

FRANK L. LEWIS, Ph.D.,
Fellow IEEE, Fellow IFAC
Professor
The Univeristy of Texas Research Institute
The University of Texas at Arlington

SHUZHI SAM GE, Ph.D.,
Fellow IEEE
Professor
Interactive Digital Media Institute
The National University of Singapore

PUBLISHED TITLES

Doubly Fed Induction Generators: Control for Wind Energy, *Edgar N. Sanchez; Riemann Ruiz-Cruz*

Optimal Networked Control Systems with MATLAB®, *Jagannathan Sarangapani; Hao Xu*

Cooperative Control of Multi-agent Systems: A Consensus Region Approach, *Zhongkui Li; Zhisheng Duan*

Nonlinear Control of Dynamic Networks, *Tengfei Liu; Zhong-Ping Jiang; David J. Hill*

Modeling and Control for Micro/Nano Devices and Systems, *Ning Xi; Mingjun Zhang; Guangyong Li*

Linear Control System Analysis and Design with MATLAB®, Sixth Edition, *Constantine H. Houpis; Stuart N. Sheldon*

Real-Time Rendering: Computer Graphics with Control Engineering, *Gabriyel Wong; Jianliang Wang*

Anti-Disturbance Control for Systems with Multiple Disturbances, *Lei Guo; Songyin Cao*

Tensor Product Model Transformation in Polytopic Model-Based Control, *Péter Baranyi; Yeung Yam; Péter Várlaki*

Fundamentals in Modeling and Control of Mobile Manipulators, *Zhijun Li; Shuzhi Sam Ge*

Optimal and Robust Scheduling for Networked Control Systems, *Stefano Longo; Tingli Su; Guido Herrmann; Phil Barber*

Advances in Missile Guidance, Control, and Estimation, *S.N. Balakrishna; Antonios Tsourdos; B.A. White*

End to End Adaptive Congestion Control in TCP/IP Networks, *Christos N. Houmkozlis; George A Rovithakis*

Robot Manipulator Control: Theory and Practice, *Frank L. Lewis; Darren M Dawson; Chaouki T. Abdallah*

Quantitative Process Control Theory, *Weidong Zhang*

Classical Feedback Control: With MATLAB® and Simulink®, Second Edition, *Boris Lurie; Paul Enright*

Intelligent Diagnosis and Prognosis of Industrial Networked Systems, *Chee Khiang Pang; Frank L. Lewis; Tong Heng Lee; Zhao Yang Dong*

Synchronization and Control of Multiagent Systems, *Dong Sun* **Subspace Learning of Neural Networks,** *Jian Cheng; Zhang Yi; Jiliu Zhou*

Reliable Control and Filtering of Linear Systems with Adaptive Mechanisms, *Guang-Hong Yang; Dan Ye*

Reinforcement Learning and Dynamic Programming Using Function Approximators, *Lucian Busoniu; Robert Babuska; Bart De Schutter; Damien Ernst*

Modeling and Control of Vibration in Mechanical Systems, *Chunling Du; Lihua Xie*

Analysis and Synthesis of Fuzzy Control Systems: A Model-Based Approach, *Gang Feng*

Lyapunov-Based Control of Robotic Systems, *Aman Behal; Warren Dixon; Darren M. Dawson; Bin Xian*

System Modeling and Control with Resource-Oriented Petri Nets, *MengChu Zhou; Naiqi Wu*

Sliding Mode Control in Electro-Mechanical Systems, Second Edition, *Vadim Utkin; Juergen Guldner; Jingxin Shi*

Autonomous Mobile Robots: Sensing, Control, Decision Making and Applications, *Shuzhi Sam Ge; Frank L. Lewis*

Linear Control Theory: Structure, Robustness, and Optimization, *Shankar P. Bhattacharyya; Aniruddha Datta; Lee H.Keel*

Optimal Control: Weakly Coupled Systems and Applications, *Zoran Gajic*

Deterministic Learning Theory for Identification, Recognition, and Control, *Cong Wang; David J. Hill*

Intelligent Systems: Modeling, Optimization, and Control, *Yung C. Shin; Myo-Taeg Lim; Dobrila Skataric; Wu-Chung Su; Vojislav Kecman*

Doubly Fed Induction Generators

CONTROL FOR WIND ENERGY

Edgar N. Sanchez

CINVESTAV, Unidad Guadalajara, Zapopan, Jalisco, Mexico

Riemann Ruiz-Cruz

ITESO, Tlaquepaque, Jalisco, Mexico

CRC Press

Taylor & Francis Group

6000 Broken Sound Parkway NW, Suite 300

Boca Raton, FL 33487-2742

CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works

Printed on acid-free paper
Version Date: 20160517

International Standard Book Number-13: 978-1-4987-4584-0 (Hardback)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Dedication

To my wife Maria de Lourdes, and our children, Zulia Mayari, Ana Maria, and Edgar Camilo

Edgar N. Sanchez

To my parents Abel and Romelia, and my brothers Urania, Neretva, and Euclides for their provided support

To Marina Flores for all her love and patience during the writing of this book

Riemann Ruiz-Cruz

Contents

List of Figures

List of Tables

Preface

Due to the vast increase in renewable energy, research on this topic is very important. In particular, the use of wind energy has spread quickly across the planet, and has proved to be very profitable. Furthermore, caused by the unpredictable wind nature, wind energy presents many challenges, such as fluctuations in the generated power due to wind variations, complications to achieve efficient coupling to the grid, and limitations produced by mechanical generator oscillations, among others. These problems have led to the development of controllers, which are able to handle such situations. The purpose of this book is to provide a detailed source of information on modeling and design of nonlinear controllers for the doubly fed induction generator (DFIG), as used in wind energy applications.

The outline of the book is as follows. In Chapter 1, the main features of the DFIG, and its advantages are explained; additionally, different publications centered on this type of generator are reviewed. The theoretical fundamentals used in this book are presented in Chapter 2. Then the DFIG mathematical model is discussed (Chapter 3). Based on this mathematical model, using the techniques of inverse discrete optimal control and sliding modes, different types of controllers are developed (Chapter 4). Subsequently, an improvement based on neural networks is proposed to add robustness in the presence of parametric variations (Chapter 5). As the last part of this book, a chapter with real-time implementation results is added (Chapter 6).

This book particularly constitutes an updated and adequate reference for people working on wind energy. It could be used by researchers and graduate students in academia as well as practitioner engineers in the industry.

Edgar N. Sanchez
Riemann Ruiz-Cruz

Guadalajara, Mexico

Acknowledgments

The authors thank CONACyT (from its name in Spanish, which stands for National Council for Science and Technology), Mexico, for financial support on project 131678. We also thank CINVESTAV-IPN (from its name in Spanish, which stands for Advanced Studies and Research Center of the National Polytechnic Institute), Mexico, particularly Rene Asomoza-Palacio, president of the CINVESTAV system, and Bernardino Castillo-Toledo, director, Guadalajara campus, during this book writing, for facilities provided to accomplish this book publication. In addition, the second author thanks the Instituto Tecnologico y de Estudios Superiores de Occidente (ITESO), Jalisco, Mexico, for allowing time to finish this book writing. Additionally, we thank Ronald G. Harley, Georgia Institute of Technology, USA; Alexander G. Loukianov, CINVESTAV, Guadalajara campus; Alma Y. Alanis, Universidad de Guadalajara, Mexico; and Fernando Ornelas-Tellez, Universidad Michoacana, Morelia, Mexico, all of whom have substantially contributed to the development of this book. We appreciate the understanding and patience of our families during this book writing.

Authors

Edgar N. Sanchez was born in 1949, in Sardinata, Colombia, South America. He earned the BSEE, majoring in Power Systems, from Universidad Industrial de Santander (UIS), Bucaramanga, Colombia, in 1971; the MSEE from CINVESTAV-IPN (Advanced Studies and Research Center of the National Polytechnic Institute), majoring in Automatic Control, Mexico City, Mexico, in 1974, and the Docteur Ingenieur degree in Automatic Control from Institut Nationale Polytechnique de Grenoble, France, in 1980. Since January 1997, he has been with CINVESTAV-IPN, Guadalajara campus, Mexico. He was granted a USA National Research Council Award as a research associate at NASA Langley Research Center, Hampton, Virginia, USA (January 1985 to March 1987). He is also a member of the Mexican National Research System (promoted to highest rank, III, in 2005), the Mexican Academy of Science, and the Mexican Academy of Engineering. He has published more than 200 technical papers in international journals and conferences, and has served as associate editor and reviewer for different international journals and conferences. He has also been a member of many international conferences (IPCs), both IEEE and IFAC. His research interest centers on neural networks and fuzzy logic as applied to automatic control systems.

Riemann Ruiz-Cruz was born in Oaxaca, Oaxaca, Mexico, in 1983. He earned the BSEE from Instituto Tecnológico de Oaxaca, Oaxaca, Mexico, in 2006; and the MSEE and D. Sc. on EE from the Advanced Studies and Research Center of the National Polytechnic Institute (CINVESTAV-IPN), Guadalajara campus, Mexico, in 2009 and 2013, respectively. Since August 2013, he has been with Instituto Tecnologico y de Estudios Superiores de Occidente (ITESO), Guadalajara, Jalisco, Mexico. He is also a member of the Mexican National Research System (Rank C). His research interests center on neural control, block control, inverse optimal control, and discrete-time sliding modes, and their applications to electrical machines and power systems.

1 Introduction

This chapter presents a brief introduction to wind energy and an overview of recent research on this topic. It also includes the book structure.

1.1 INTRODUCTION AND OVERVIEW OF RECENT RESEARCH

Wind energy, as an alternative to fossil fuels, is plentiful, renewable, widely distributed, clean, and produces no greenhouse gas emissions. Wind power is the conversion of wind energy into a useful form of energy, such as using wind turbines to produce electricity, windmills for mechanical power, wind pumps for water pumping or drainage, or sails to propel ships.

Without doubt wind power has become a pillar of the energy systems in many countries and is recognized as a reliable and affordable source of electricity. In the year 2012, the worldwide wind capacity reached 282,275 Megawatt (MW), after 236,749 MW in 2011, 196,944 MW in 2010, and 159,742 MW in 2009. The market for new wind turbines reached a new record: 44,609 MW was installed in 2012, an increase of 12% compared with 2011 when 39,805 MW was erected. The contribution of wind power to the energy supply has reached a substantial share even on the global level: All wind turbines installed around the globe by the end of 2011 contribute potentially 580 Terawattshours to the worldwide electricity supply, more than 3% of the global electricity demand [11].

In the year 2012, 100 countries were identified where wind energy was used for electricity generation. The 100th entrant on the list is Iceland, a country which has already almost 100% of its energy supply coming from renewable energy. Forty-six countries installed new turbines, four less than in the previous year and after even 52 in 2010 [11].

Large-scale wind farms are connected to the electric power transmission network; smaller facilities are used to provide electricity to isolated locations. Utility companies increasingly buy back surplus electricity produced by small domestic turbines. The construction of wind farms is not universally welcomed because of their visual impact; however, any effects on the environment from wind power are generally less problematic than those of any other power source [10].

The intermittency of wind seldom creates problems when using wind power to supply a low proportion of total demand, but as the proportion rises, increased costs, a need to upgrade the grid, and a lowered ability to supplant conventional production may occur. Power management techniques, such as exporting and importing power to neighboring areas or reducing demand when wind production is low, can mitigate these problems [10].

A wind farm is a group of wind turbines used for production of electric power. A large wind farm may consist of several hundred individual wind turbines, and cover an extended area of hundreds of square meters; the land between the turbines may be used for

agricultural or other purposes. A wind farm may also be located offshore [10].

A very good indicator for the vitality of the market development is the average growth rate. The growth rate is the relation between the new installed wind power capacity and the installed capacity of the previous year. After an average growth of 30% in the past decade, the growth has decreased substantially in the past three years: In 2012, the global growth went down to 19.1%, the lowest rate in two decades [11].

In 2012, Latinoamerica became the most dynamic continent for wind power investment, mainly due to Brazil and Mexico. The growth rate of 56% was the highest worldwide. Eight Latin American countries installed new wind turbines in the year 2012: Brazil (1077 MW), Mexico (419 MW), Argentina (104 MW), Puerto Rico (125 MW), Nicaragua (40 MW), Uruguay (24 MW), Venezuela (30 MW), and Ecuador (17 MW).

Mexico is number 24 in the ranking of countries with wind power installed in the world and rank second in Latinoamerica. By the end of 2010, Mexico had installed 521 MW of wind energy [10], and by the end of 2012, 1348 MW [11].

Due to the need to reduce emissions, the use and development of clean energies have increased in recent years around the world. So, it is necessary to develop wind energy controllers in order to maximize the extraction of wind energy and to guarantee the quality of this energy.

The doubly fed induction generator (DFIG), based wind turbine has been used widely for interconnection to the grid. As a variable-speed wind generator, the DFIG has a number of advantages compared to fixed-speed generators, including that the machine can operate in sub-synchronous, synchronous, as well as super-synchronous mode, decoupled control of active and reactive power, better energy capture, mechanical stress reduction, and low cost with the development of the power electronics converters [50]. The conventional topology, the bi-directional buck-boost DC-DC converter is connected between the fixed DC Link voltage. The rotor side converter (RSC) is connected to the rotor in order to control the active and reactive power of the DFIG stator. The grid side converter (GSC) is connected to the grid and controls the DC Link voltage [61, 65].

There are already published books that address wind energy, including the DFIG. In [9], the objective of that book is to provide a basic understanding of wind generation systems modeling, including both the mechanical and electrical systems, and to examine control philosophies and schemes that enable reliable, secure, and cost-effective operation. In [15, 16, 83], the basic electrical concepts to understand the operation of the DFIG are presented. Different control schemes for generator control are presented, too. These books are good references, because they explain simply and clearly the electrical and mechanical behavior of the DFIG, but details of controller design procedure are not reported, and the results are illustrated through simulations. In [24], different controllers for the complete DFIG scheme are presented, involving the RSC and GSC; most of these controllers are linear and not present details. The major emphasis of the book is targeted on multi-machine DFIG applications. Abad et al. [2] are focused on the modeling and control of the doubly fed induction machine (DFIM)-based wind turbines; this book is a good reference, because the control techniques studied are the standard solutions used by wind turbine

manufacturers, including the last developments oriented to improve the behavior of high power wind turbines, as well as control, and hardware-based solutions to address different faulty scenarios of the grid. Moreover, almost all controllers proposed are linear and can be improved using nonlinear ones. A quite practical reference is [3], where different control techniques are studied and analyzed by means of simulations in MATLAB/Simulink®[1]. It is worth noting that all the books mentioned above are based on mathematical models in continuous time and the results are presented by means of simulations.

A typical configuration of a DFIG wind turbine is composed of a wound-rotor induction generator with slip rings used to take current into or out of the rotor winding, and variable-speed operation is obtained by injecting a controllable voltage into the rotor at slip frequency. The rotor winding is fed through a variable-frequency power converter, typically based on two AC/DC insulated-gate bipolar transistor (IGBT)-based voltage source converters, namely, RSC and GSC, linked by a DC bus. This configuration in known as variable frequency AC/DC/AC converter (VFC), which only needs to handle a fraction (25%-30%) of the total DFIG power to achieve full control of the generator. The GSC is connected to the stator terminals directly or through a step-up transformer. Both RSC and GSC are four-quadrant converters which allow bi-directional power flow; different techniques have been proposed for this configuration control. In [18], a complete overview of the control systems for the DFIG based on different configurations is presented, and the importance of this type of generator in wind energy applications is highlighted. In [46], a steady-state stability analysis for the effect of rotor excitation voltage of the DFIG and experimental results are given to demonstrate the effectiveness of the theoretical analysis.

Vector control (VC) and direct power control (DPC) are the common techniques for induction machines [1, 55, 62, 67], and the proportional integral (PI) controller based on decoupling is the traditional control scheme for the DFIG [13, 63, 66]; these PI controllers are designed for a limited operation region. Additionally, recent research has suggested that the control signals generated by the d-q axis PI controllers do not contribute correctly in terms of the decoupled d-q control objectives [45]. In [41], an exact feedback linearization technique is applied to implement a non linear controller; this nonlinear technique is based on the DFIG mathematical model; then the controller performance is affected by parameter variations. Additionally, in all the mentioned papers the controller performance is only validated by simulations.

Sliding modes (SM) is a technique which adds robustness to controller performance in the presence of disturbances and parameter variations [78, 85]. In [4, 14, 51, 87], distinct SM schemes are applied for DFIG controllers in different operation regions, and simulation results validate the robustness of this control technique. Even if SM is robust, there is not guarantee of optimal tracking; in addition, the controllers are designed in continuous time and it will be difficult to implement them in real time.

The DFIG operates in continuous time and can be represented using ordinary differential equations; however, for real time implementations, it is more convenient to use

discrete time controllers. Besides, there are two advantages to working in a discrete time framework: 1) appropriate technology can be used to implement digital controllers rather than analog ones; 2) the synthesized controller is directly implementable in a digital processor. Additionally, it is not guaranteed that a continuous time control scheme preserves its properties when implemented in real time; even worse, it is known that continuous time schemes could become unstable after sampling [54]. Considering these facts, in this book we use discrete time representations for the DFIG model and the respective controllers. In [70], a discrete time block control scheme with sliding modes is used to design a controller for rotor speed tracking and to keep the stator power factor constant by means of the stator reactive power output; however, this paper only presents the RSC controller, and the simulation results neglect the IGBT's dynamics. There are few papers to show DFIG controllers real time performance; in [30], a real time implementation is achieved using PI controllers with fuzzy logic on a commercial microcontroller; however, the operation region is limited due to PI control usage. In [52], a real time implementation of a torque controller for a DFIG is presented; in this paper the sliding mode algorithm is used.

In [88], a predictive direct power control strategy for a DFIG is presented. The method predicts the DFIG's stator active and reactive power variations within a fixed sampling period, which is used to directly calculate the required rotor voltage to eliminate the power errors at the end of the next sampling period. In [82], an optimal neurocontroller for the conventional automatic voltage regulator (AVR) and the turbine governor for a turbogenerator connected to the power grid are discussed. It shows that the neurocontroller is robust especially when the system conditions and configuration change. In [86], a heuristic dynamic programming iteration algorithm is proposed to solve the optimal tracking control problem for a class of nonlinear discrete time systems with time delays, where a neural network is used to approximate the performance index function and compute the optimal control policy. In [44], a neural network is used to control a grid-connected rectifier/inverter. The performance of this neural-network controller is studied under typical vector control conditions and compared against conventional vector control methods, which demonstrates that the neural vector control strategy is effective. Even in switching environments, the neural vector controller shows strong ability to trace rapidly changing reference commands, to tolerate system disturbances, and to satisfy control requirements for a faulted power system.

In this book, emphasis is given to the detailed synthesis of control algorithms for a DFIG, using novel nonlinear schemes to provide robustness against unmodeled disturbances. All controllers discussed in this book are designed in discrete time and validated through simulations and real time implementations.

1.2 BOOK STRUCTURE

In this book, the authors propose different control schemes based on sliding modes, neural network, and inverse optimal control, for a DFIG connected to an infinity bus.

The control schemes can be classified in two categories:

1) controllers based on the plant mathematical model,

2) and controllers based on a neural network model.

The outline of this book is as follows.

Chapter 2 introduces mathematical preliminaries used in the development of this book. The control scheme developments included in this book are based on the mathematical theory described in this chapter. In the first section, information on control blocks is presented; this transformation facilitates the controller synthesis of nonlinear systems. Then, an overview of the sliding modes fundamentals, and their equivalent in discrete time, is addressed in a general way. After that, the framework of the novel inverse optimal control is briefly discussed, which is used to obtain optimal control inputs to reach desired control objectives. Due to neural network robustness to uncertainties, the background of recurrent high-order neural networks in discrete time for the modeling of nonlinear systems is introduced; the respective employed learning algorithm is based on the extended Kalman filter. The neural network is used to approximate the plant mathematical model, which can be used for obtaining controllers robust in presence of parameter variations and external disturbances. Additionally, the theoretical framework of an heuristic optimization algorithm is introduced; this algorithm can be used to optimize the performance of controllers through the search of specific parameters.

Chapter 3 presents a description of wind energy system configuration and the discrete time mathematical model development for the configuration to be used. In this chapter, a brief discussion about wind energy is presented, which is the most promising green and clean energy, does not generate greenhouse gases, and is abundant and seemingly endless. One disadvantage is that the geographic place where you can take energy from wind must present a continuous flow of air for much of the year. Studies have identified places where wind is plentiful and its possible utilization. There are different types of generators that take advantage of wind energy. Each configuration has its advantages, but the typical configuration used for wind power is to use a DFIG. This generator can be connected directly to the grid, and allows the variation of the rotor mechanical speed, allowing a wider range of speeds where the generator can continue generating electricity. In this book the discrete time mathematical model of a DFIG is used.

Chapter 4 discusses a detailed development of different controllers for a DFIG. The controllers are synthesized to track a desired electric torque (τ_e), while adjusting the respective reactive power (Q_s) in order to obtain a desired power factor (f_{ps}) at the generator stator terminals; this synthesis is possible taking advantage of the generators ability to allow controlling active power and reactive power separately. This chapter is divided into two parts. In the first part, a discrete time sliding mode controller is used. The algorithm is applied for both RSC and GSC modules of the DFIG. In the second part, the control scheme is based on the inverse optimal control approach. The performance of these controllers are tested by means of simulations using MATLAB®[2].

Chapter 5 introduces a detailed reformulation of the controllers explained in the previous chapter, using recurrent high-order neural networks (RHONN). Assuming that

the DFIG operation is affected by both time-varying parameters and external disturbances, in this chapter two hybrid control schemes are presented. The first one is the neural sliding mode controller, which uses a neural identifier to obtain a mathematical model of the DFIG and the DC Link; then the sliding mode algorithm is applied based on this neural mathematical model. Next, a second control scheme is presented, in which an inverse optimal controller, based on the DFIG neural model, is applied. All validations of this chapter are presented through MATLAB® simulations.

Chapter 6. In this chapter, real time implementation results are presented for the controllers previously explained. These implementations are done using a DFIG prototype, whose description is included. Supported by the simulations and a prototype of a DFIG, the real time implementation was achieved for multiple controllers designed in this book. All the controllers are tested under the same conditions; the different controllers, performances are compared by means of statistical measures as mean values and standard deviation of the error.

Additionally, two appendices are included at the end of this book. Appendix A covers the use of particle swarm optimization (PSO) to determine parameters of the above-mentioned controllers. As explained, PSO is used to determine a matrix required by inverse optimal control, and to define matrices needed by the extended Kalman filter (EKF) neural networks training. In Appendix B, the DFIG and DC Link mathematical model development is detailed. Based on [38], the entire process of modeling an induction motor is developed; then based on this procedure, we derive the full mathematical model of a DFIG. Using the Euler approximation, we obtain the DFIG discrete time mathematical model, which is used as the basis for this book.

1.3 NOTATION

Through out this book, we use the following notation:

Symbol	Description
$\|\bullet\|$	Euclidean norm for vectors and any adequate norm for matrices
A	Swept area of wind turbine
C_p	Wind turbine power coefficient
C	DC Link capacitance
$H_i \in \Re^{L_i \times m}$	Derivative of the neural network output
H	Moment of inertia
i_{ds}	Stator current in the d axis
i_{qs}	Stator current in the q axis
i_{dr}	Rotor current in the d axis
i_{qr}	Rotor current in the q axis

i_{dg}	GSC current in the *d* axis
i_{qg}	GSC current in the *q* axis
k_i	Control gain of the *i*th block
$k \in 0 \cup \mathbb{Z}^+$	Sampling step
$K \in \Re^{L_i \times m}$	Kalman gain matrix
$l(z)$	Positive semidefinite function
$L_i \in \Re$	Number of neural network weights
λ	Tip-speed ratio
$M_i \in \Re$	Number of high-order connections
$n_i \in \Re$	Dimension of the *i*th block
$\omega_i \in \Re^L$	*i*th neural network weight vector estimation
$\boldsymbol{\omega_i^* \in \Re^L}$	*i*th neural network ideal weight vector
ω_b	Synchrony frequency
ω_r	DFIG rotor speed
P	Memory set of particle swarm
P_{air}	Power in the airflow
$P \in \Re^{n \times n}$	Positive definite
$P_i \in \Re^{L_i \times L_i}$	Associated prediction error covariance matrix
$Q_i \in \Re^{L_i \times L_i}$	Associated state noise covariance matrix
$\rho \in \Re^m$	Neural network external input
R	Uniformly distributed random variables
$r \in \Re$	Number of blocks
$R_i \in \Re^{m \times m}$	Associated measurement noise covariance matrix
r_s	Stator resistance per phase
r_r	Rotor resistance per phase
ρ	Air density
R	Wind turbine radius
R_g	GSC resistance of the three phase lines *a, b, c*
$S(\bullet)$	Sigmoid function
Sw	Particle swarm

t_s	Sampling time
τ_e	Electromagnetic torque (*pu*)
τ_m	Mechanical torque
$u \in \Re^m$	Control action
$u^*(k)$	Optimal control law
v	Wind speed
v_{ds}	Stator voltage in the *d* axis
v_{qs}	Stator voltage in the *q* axis
v_{dr}	Rotor voltage in the *d* axis
v_{qr}	Rotor voltage in the *q* axis
v_{dc}	DC Link voltage
v_{dgs}	GSC voltage in the *d* axis
v_{qgs}	GSC voltage in the *q* axis
$V(\bullet)$	Lyapunov function
$\Delta V(\bullet)$	Lyapunov difference
w_i	Weights vector
$x \in \Re^n$	Plant state
$x_i \in \Re$	Component of particle solution
$\hat{\boldsymbol{x}} \in \boldsymbol{\Re^n}$	Neural network state
$x(k) \in \mathbb{R}^n$	State of the system at time $k \in \mathbb{Z}^+ = \{0,1,2,\ldots\}$
X_s	Stator self-reactance per phase
X_r	Rotor self-reactance per phase
X_m	Magnetization reactance
X_l	GSC reactance of three phase lines *a, b, c* (*pu*)
y	Outputs vector
$\hat{y}$	Neural network output
$z_i \in \Re^{n_r}$	State transformation of the *i*th block
$z_i \in \Re^{M_i}$	High-order terms

NOTES

[1] MATLAB/Simulink® is a trademark of MathWorks.

[2] MATLAB/Simulink® is a trademark of MathWorks.

2 Mathematical Preliminaries

This chapter introduces mathematical preliminaries used in the development of this book.

2.1 BLOCK CONTROL

The robust control synthesis for nonlinear dynamical systems is an interesting research subject. A fruitful and relatively simple approach to solve this problem, when dealing with multi-variable non linear uncertainties, is based on the use of the variable structure control approach with sliding mode [79].

2.1.1 BLOCK REPRESENTATION OF A CLASS OF NONLINEAR SYSTEMS

In order to illustrate the potential of block decomposition, consider the following system subject to uncertainty:

$$\dot{x} = f(x,t) + B(x,t)\,u + g(x,t), \tag{2.1}$$

where $x \in X \subset \Re^n$ is the state vector, $u \in U \subset \Re^m$ is the control vector to be bounded by

$$u = (u_1, \ldots, u_m)^T, \ \|u_i\| \le U_0 \text{ with } U_0 > 0, \ i = 1, \cdots, m. \tag{2.2}$$

The unknown mapping $g(x, t)$ characterizes external disturbances and parameter variations which should not affect the feedback systems. It is assumed that the vector fields $f(x, t)$, $g(x, t)$ and the columns of $B(x, t)$ are smooth and bounded mappings of class $C^{\infty}_{[0,\infty]}$, $f(0,t) = 0$, and rank $B(x, t) = m$ for all $x \in X$ and $t \ge 0$ [47].

The essential feature of the proposed method is the conversion of the system (2.1) to the block control (BC) form consisting of r blocks:

$$\begin{aligned}
\dot{x}_1 &= f_1(x_1,t) + B_1(x_1,t)\,x_2 + g_1(x_1,t),\\
\dot{x}_i &= f_i(\overline{x}_i,t) + B_i(\overline{x}_i,t)\,x_{i+1} + g_i(\overline{x}_i,t),\\
\dot{x}_r &= f_r(\overline{x}_r,t) + B_r(\overline{x}_r,t)\,u + g_r(\overline{x}_r,t),
\end{aligned} \tag{2.3}$$

where the vector x is decomposed as $x = (x_1, x_2, \ldots, x_r, x_{r+1})^T$, $\overline{x}_i = (x_1, \ldots, x_i)^T$, $i = 2, \ldots, r$, x_i is a $n_j \times 1$ vector, and the indices $(n_1, n_2, \ldots, n_r)$ define the structure of the system and satisfy the following relation:

$$n_1 \leq n_2 \leq \cdots \leq n_r \leq m \text{ and } \sum_{i=1}^{r} n_i = n. \tag{2.4}$$

The matrix B_i, multiplying the fictitious x_{k+1} in each ith block of (2.3), has full rank, that is,

$$rank\, B_i = n_i \; \forall x \in X \subset \Re^n \text{ and } t \in [0, \infty), \; i = 1, \ldots, r. \tag{2.5}$$

The procedure of reducing system (2.1) to the BC form (2.3) based on the integral transformation method is presented in [47, 48].

Let us first consider the plant with the structure

$$n_1 < n_2 < \cdots < n_r < m. \tag{2.6}$$

2.1.2 BLOCK RECURSIVE TRANSFORMATION

The following assumption on bounds for the unknown terms in (2.1) is stated. There exist positive constants $\bar{q}_{i,j}$ and $\bar{d}_i$ such that

$$\begin{aligned}
\|g_1(\bar{x}_1, t)\| &\leq \bar{q}_{11} \|x_1\| + \bar{d}_1, \\
\|g_2(\bar{x}_2, t)\| &\leq \bar{q}_{21} \|x_1\| + \bar{q}_{22} \|x_2\| + \bar{d}_2, \\
\|g_i(\bar{x}_i, t)\| &\leq \sum_{j=1}^{i} \bar{q}_{ij} \|x_j\| + \bar{d}_i, \; i = 3, \ldots, r-1.
\end{aligned}$$

Considering structure (2.6), the following recursive transformation is introduced:

$$\begin{aligned}
z_1 = x_1 &:= \Phi_1(x_1, t) \\
z_2 = \widetilde{B}_2(\bar{x}_1, t)\, x_2 + \begin{bmatrix} f_1(x_1, t) + k_1 \Phi_1(x_1, t) \\ 0 \end{bmatrix} &:\equiv \Phi_2(\bar{x}_2, t) \\
z_{i+1} = \widetilde{B}_{i+1}(\bar{x}_i, t)\, x_{i+1} + \begin{bmatrix} f_i(x_i, t) + k_i \Phi_i(x_i, t) \\ 0 \end{bmatrix} &:\equiv \Phi_{i+1}(\bar{x}_{i+1}, t) \\
i = 3, \ldots, r-1 &
\end{aligned} \tag{2.7}$$

where z_i is a $n_i \times 1$ new variables vector, $k_i > 0$, $\widetilde{B}_{i+1} = \begin{bmatrix} \bar{B}_i \\ E_{i,2} \end{bmatrix}$, $E_{i,2} = \begin{bmatrix} 0 \, I_{n_{i+1}-n_i} \end{bmatrix}$, and $E_{i,2} \in \Re^{(n_{i+1}-n_i) \times n_{i+1}}$, $I_{n_{i+1}-n_i}$ is the identity matrix.

Transformation (2.7) reduces system (2.1) to the following desired form:

$$\begin{aligned}
\dot{z}_1 &= -k_1 z_1 + E_{11} z_2 + \bar{g}_1(z_1, t) \\
\dot{z}_i &= -k_i z_i + E_{i,1} z_{i+1} + \bar{g}_i(\bar{z}_i, t), \; i = 2, \ldots, r-1 \\
\dot{z}_r &= \bar{f}_r(z, t) + \bar{B}_r(z, t)\, u + \bar{g}_r(z, t),
\end{aligned} \tag{2.8}$$

where $z = (z_1, \ldots, z_r)^T$, $\bar{f}_r(z, x_{r+1}, t)$ is a bounded function, rank $\bar{B}_r = n_1$, and $\bar{B}_r = \tilde{B}_{r-1} B_r$.

2.2 SLIDING MODES

Most sliding mode approaches are based on finite-dimensional continuous time models and lead to discontinuous control action. Once a dynamic system is in sliding mode, its trajectory is confined to a manifold in the state space. Generally this condition of the system may only be achieved by discontinuous control, switching at theoretically infinite frequency. When challenged with the task of implementing sliding mode control in a practical system, the control engineer has two options:

1. Direct analogue implementation of a discontinuous control law with a very fast switching device
2. Discrete implementation of sliding mode control

The first method is only suitable for systems with a voltage input allowing the use of analogue switching devices. Most systems are usually based on a discrete microcontroller-based implementation. However, a discontinuous control designed for a continuous time system model would lead to chattering when implemented without modifications in discrete time with a finite sampling rate. This discretization chattering is due to the fact that switching frequency is limited to the sampling rate; however, correct implementation of sliding mode control requires infinite switching frequency. The following example illustrates the difference between ideal continuous time sliding mode and direct discrete implementation with discretization chattering.

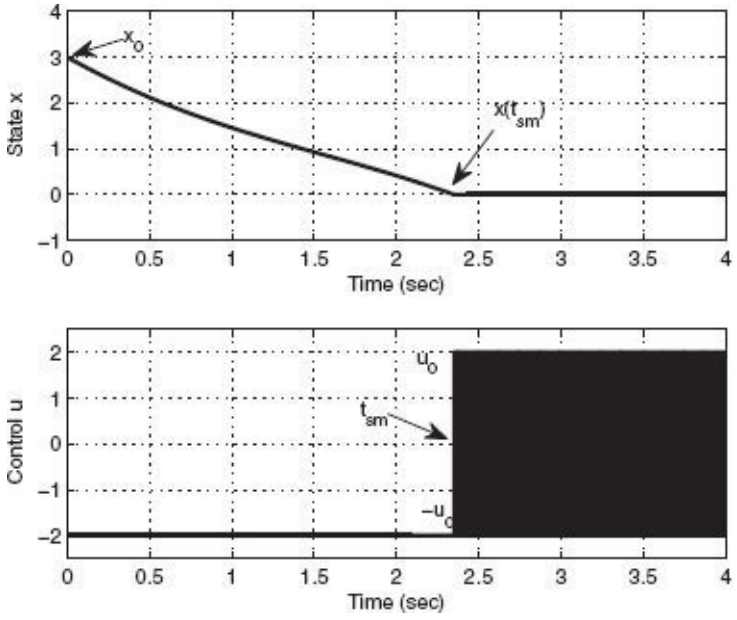

Figure 2.1 Ideal sliding mode.

Figure 2.1 shows the ideal sliding mode for a first-order example system, considering an achieved via direct, analog implementation of a discontinuous control law with infinitely fast switching.

In Figure 2.2, the direct implementation of sliding mode control in discrete time is

shown. Sampling instances are marked with small circles. Control uk may only be switched at sampling instances, resulting in discretization chatter in the motion trajectory after reaching the vicinity of the sliding manifold at $t_{sm} = k_{sm}\Delta t$.

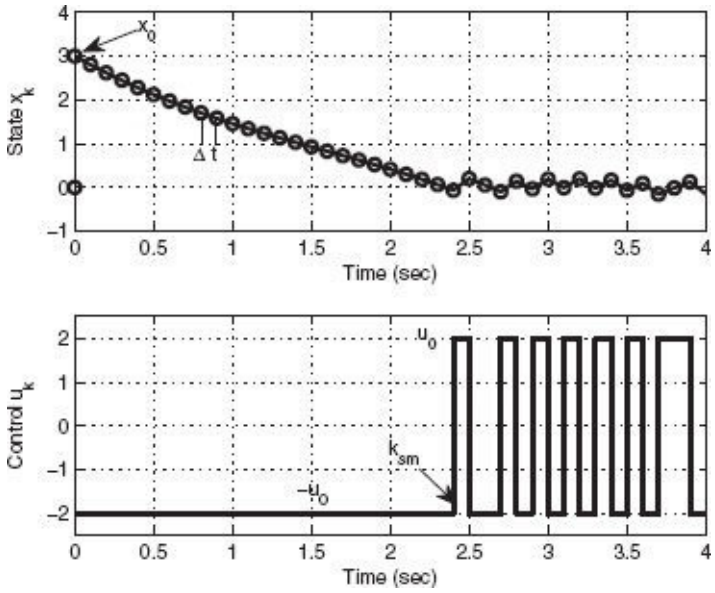

Figure 2.2 Direct implementation of sliding mode control in discrete time.

Example 1 *Consider a first-order system modeled in continuous time as*

$$\dot{x} = g(x(t)) + u(t), \tag{2.9}$$

with state x(t), bounded dynamics $|g(x)| \leq \bar{g}$, *and control input u(t). To enforce sliding mode on a manifold*

$$\sigma = \{x : x(t) = 0\}. \tag{2.10}$$

A discontinuous control law is designed as

$$u(t) = -u_0 \, sign \, x(t), \tag{2.11}$$

with available control resources $u_0 > \bar{g}$. *The usual Lyapunov-based stability analysis examines examines*

$$V = \tfrac{1}{2}x^2(t), \tag{2.12}$$

along the trajectories of system (2.9) with control (2.11), leading to

$$\begin{aligned} \dot{V}(t) &= x(t)\left(g(x(t)) - u_0 \, sign \, x(t)\right) \\ &\leq -|x(t)|\left(u_0 - \bar{g}\right), \end{aligned} \tag{2.13}$$

which testifies to convergence to the manifold (2.10) within finite time. An example

trajectory is shown in Figure 2.1, with $g(t) = sin(t)$ and $u_0 = 2$, starting from initial conditions $x(t = 0) = 3$. At t_{sm} the system reaches the sliding manifold $x = 0$. Thereafter, the motion trajectory is invariantly confined to the manifold (2.10) via discontinuously switching control, illustrated in the bottom diagram of Figure 2.1 by a black rectangle.

A direct discrete implementation with sampling time Δt would result in

$$\begin{aligned} x_{k+1} = \; & x_k + (g_k + u_k)\,\Delta t, \\ & u_k = -u_0\, sign\, x_k, \\ & k = 1, 2, \ldots, \end{aligned} \tag{2.14}$$

where subscript k denotes the sampling points, e.g., the system state x_k at time instance $t_k = k\Delta t$. The motion trajectory may not reach the manifold $x = 0$ because control u_k is only calculated at sampling points k, i.e., the switching frequency is limited by the sampling rate $1/\Delta t$. During the sampling interval Δt, the control is constant and the system behaves like an open-loop system [37].

The example with exaggerated large sampling time $\Delta t = 0.1$ sec as depicted in Figure 2.2 illustrates the need to develop a discrete time sliding mode algorithm rather than implementing (2.14).

It should be noted that increasing the sampling rate decreases the amplitude of the discretization chatter and increases its frequency but may not eliminate this discrete time phenomenon unless $\Delta t \longrightarrow 0$. Moreover, the sampling rate of a control system should correspond to the fastest dynamics of the system instead of wasting computational power for the control algorithm.

2.2.1 DISCRETE TIME SLIDING MODES

Before developing the concept of discrete time sliding mode, let us revisit the principle of sliding mode in continuous time systems with ideal discontinuous control from an engineering viewpoint. A more mathematical treatment may be found in [79] or in [21]. Rewrite (2.9) as a general continuous time system

$$\dot{x} = f(x, u, t) \tag{2.15}$$

with a discontinuous scalar control law

$$u = \begin{cases} u_0 \; if \; s(x) \geq 0 \\ -u_0 \; if \; s(x) < 0 \end{cases} \tag{2.16}$$

and sliding mode in a manifold $s(x) = 0$.

Note the following observations characterizing the nature of sliding mode systems:

The time interval between the initial point $t = 0$ and the reaching of the sliding

manifold $\sigma = \{x : s(x) = 0\}$ at t_{sm} is finite, in contrast to systems with a continuous control law that exhibit asymptotic convergence to any manifold consisting of state trajectories.

Once the system is in sliding mode for all $t \geq t_{sm}$, its trajectory motion is confined to the manifold $\sigma = \{x : s(x) = 0\}$ and the order of the closed-loop system dynamics is less than the order of the original uncontrolled system. After sliding mode has started at t_{sm}, the system trajectory cannot be backtracked beyond the manifold $\sigma = \{x : s(x) = 0\}$ like in systems without discontinuities. In other words, at any point $t_0 \geq t_{sm}$, it is not possible to determine the time t_{sm} or to reverse calculate the trajectory for $t < t_{sm}$ based on information of the system state at t_0.

However, during both time intervals before and after reaching the sliding manifold, the state trajectories are continuous functions of time, and the relation between two values of the state at the ends of a finite time interval $t = [t_0, t_0 + \Delta t]$ may be found by solving (2.15) as

$$\boldsymbol{x}\left(t_0 + \Delta t\right) = \boldsymbol{F}\left(\boldsymbol{x}\left(t_0\right)\right), \tag{2.17}$$

where $F(x(t))$ is a continuous function as well. When derived for each sampling point $t_k = k\Delta t$, $k = 1, 2, \ldots$, equation (2.17) is the discrete time representation of the continuous time (2.15),

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}\left(\boldsymbol{x}_k\right), \boldsymbol{x}_k = \boldsymbol{x}\left(k\Delta t\right). \tag{2.18}$$

Starting from time t_{sm}, the state trajectory belongs to the sliding manifold with $s(x(t)) = 0$, or for a $k_{sm} \geq t_{sm}/\Delta t$,

$$\boldsymbol{s}\left(\boldsymbol{x}_k\right) = 0, \quad \left(\forall \boldsymbol{k} \geq \boldsymbol{k}_{sm}\right). \tag{2.19}$$

It seems reasonable to call this motion *sliding mode in discrete time* or *discrete time sliding mode*. Note that the right-hand side of the system equation with discrete time sliding mode is a continuous state function.

So far, we have generated a discrete time description of a continuous time sliding mode system. Next, we need to derive a discrete time control law which may generate sliding mode in a discrete time system. Let us return to the example (2.15) and suppose that, for any constant control input u and any initial condition $x(0)$, the solution to (2.15) may be found in closed form,

$$\boldsymbol{x}\left(t\right) = \boldsymbol{F}\left(\boldsymbol{x}\left(0\right), \boldsymbol{u}\right). \tag{2.20}$$

Now also assume that control u may be chosen arbitrarily. With (2.20), the below

procedure follows:

1. At $t = 0$, select constant $u(x(t = 0), \Delta t)$ for a given time interval Δt such that $s(x(t = \Delta t)) = 0$.
2. At $t = \Delta t$, select different constant $u(x(t = \Delta t), \Delta t)$ such that $s(x(t = 2\Delta t)) = 0$.
3. In general, for each $k = 0,1,2,\ldots$, at $t = k\Delta t$ choose constant $u(x_k, \Delta t)$ such that $s(x_{k+1}) = 0$.

In other words, at each sampling point k, select u_k such that this control input is constant during the next sampling interval Δt, and will achieve $s(x_{k+1}) = 0$ at the next sampling point $(k + 1)$. During the sampling interval, state $x(t)$ where $k\Delta t < t < (k + 1)\Delta t$ may not belong to the manifold, $s(x(t)) \neq 0$ is possible for $k\Delta t < t < (k + 1)\Delta t$. However, the discrete time system

$$\begin{aligned} x_{k+1} &= F(x_k, u_k) \\ u_k &= u(x_k) \end{aligned} \tag{2.21}$$

hits the sliding manifold at each sampling point, $s(x_{k+1}) = 0\ \forall k = 0,1,2,\ldots$.

Since $F(x(0), u)$ tends to $x(0)$ as $\Delta t \rightarrow 0$, the function $u(x(0),\Delta t)$ may exceed the available control resources u_0. As a result, the bounded control shown in the lower diagram of Figure 2.3 steers state x_k to zero only after a finite number of steps k_{sm}. Thus, the manifold is reached after a finite time interval $t_{sm} = k_{sm}\Delta t$, and, thereafter, the state x_k remains on the manifold. In analogy to continuous time systems, this motion may be called *discrete time sliding mode*. Note that sliding mode is generated in the discrete time system with control $-u_0 \leq u \leq u_0$ as a continuous function of the state x_k and is piecewise constant during the sampling interval.

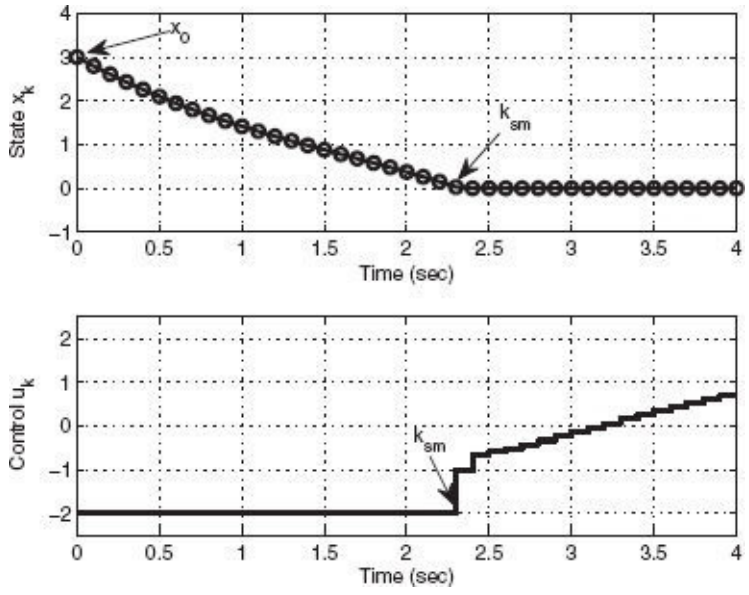

Figure 2.3 Proper implementation of sliding mode control in discrete time.

Figure 2.3 shows the proper implementation of sliding mode control in discrete time. Sampling instances are marked with small circles. Control u_k is selected as $-u_0 \le u_k \le u_0$ at each sampling instance to achieve $s(x_{k+1}) = 0$ as quickly as possible in accordance with the available control resources, resulting in chatter-free motion after reaching the sliding manifold at $t_{sm} = k_{sm}\Delta t$.

This first-order example clarifies the definition of the term *discrete time sliding mode* introduced by [21] for an arbitrary finite-dimensional discrete time system.

Definition 1 ***Discrete-Time Sliding Mode***. *In the discrete time dynamic system*

$$x_{k+1} = F(x_k, u_k) \quad (x \to \Re^n,\ u \in \Re^m,\ m \le n) \tag{2.22}$$

discrete time sliding mode takes place on a subset Σ *of the manifold* $\sigma = \{x : s(x) = 0\}$, $s \in \Re^m$, *if there exists an open neighborhood* ε *of this subset such that for each* $x \in \varepsilon$ *it follows that* $s(F(x_{k+1})) \in \Sigma$.

In contrast to continuous time systems, sliding mode may arise in discrete time systems with a continuous function in the right-hand side of the closed-loop system equation. Nevertheless, the aforementioned characteristics of continuous time sliding mode have been transferred to discrete time sliding mode. The mathematical implications in terms of group theory may be found in [21, 79].

2.2.2 DISCRETE TIME SYSTEMS WITH KNOWN PARAMETERS

[] This section deals with discrete time sliding mode control for time-invariant continuous time plants. Let us assume that a sliding mode manifold is linear for an nth-order discrete time system $x_{k+1} = F(x_k)$, $s_k = Cx_k$, $C \in \Re^{m\times n}$ with m control inputs. According to Definition 1, the sliding mode existence condition is of the form

$$s_{k+1} = CF(x_k) \tag{2.23}$$

for any $x_k \in \varepsilon$. To design a discrete time sliding mode control law based on condition (2.23), consider the continuous time representation of the linear time-invariant system

$$\dot{x}(t) = Ax(t) + Bu(t) + Dr(t) \tag{2.24}$$

with state vector $x(t) \in \Re^n$, control $u(t) \in \Re^m$, reference input $r(t)$, and constant system matrices A, B, and D. Transformation to discrete time with sampling interval Δt yields

$$x_{k+1} = A^* x_k + B^* u_k + D^* r_k, \tag{2.25}$$

where

$$A^* = e^{A\Delta t}, \quad B^* = \int_0^{\Delta t} e^{A(\Delta t - t)} B d\tau, \quad D^* = \int_0^{\Delta t} e^{A(\Delta t - t)} D d\tau, \tag{2.26}$$

and the reference input $r(t)$ is assumed to be constant during the sampling interval Δt. In accordance with (2.23), discrete time sliding mode exists if the matrix CB^* has an inverse and the control u_k is designed as the solution of

$$s_{k+1} = CA^* x_k + CD^* r_k + CB^* u_k = 0. \tag{2.27}$$

In other words, control u_k should be chosen as

$$u_k = -(CB^*)^{-1} \left(CA^* x_k + CD^* r_k\right). \tag{2.28}$$

By analogy with continuous time systems, control law (2.28), yielding motion in the manifold $s = 0$, will be called *equivalent control*. To reveal the structure of $u_{k_{eq}}$, let us represent it as the sum of two linear functions:

$$u_{k_{eq}} = -(CB^*)^{-1} s_k - (CB^*)^{-1} \left((CA^* - C)\, x_k + CD^* r_k\right) \tag{2.29}$$

and

$$s_{k+1} = s_k + (CA^* - C)\, x_k + CD^* r_k + CB^* u_k. \tag{2.30}$$

$u_{k_{eq}}$ may exceed the available control resources with $\Delta t \longrightarrow 0$ for initial $s_k \neq 0$, since $(CB^*)^{-1} \longrightarrow \infty$ means that $(CB^*)^{-1}\,(CA^* - C)$ and $(CB^*)^{-1} CD^*$ take finite values. Hence the real-life bounds for control u_k should be taken into account.

Suppose that the control can vary within $\|u_k\| \leq u_0$ and the available control resources are such that

$$\left\|(CB^*)^{-1}\right\| \left\|(CA^* - C)\, x_k + CD^* r_k\right\| < u_0. \tag{2.31}$$

Note that the control resources are sufficient to stabilize the system. The control

$$u_k = \begin{cases} u_{k_{eq}} & \text{for } \|u_{k_{eq}}\| \leq u_0 \\ u_0 \dfrac{u_{k_{eq}}}{\|u_{k_{eq}}\|} & \text{for } \|u_{k_{eq}}\| > u_0 \end{cases} \tag{2.32}$$

complies with the bounds on the control resources. As shown above, $u_k = u_{k_{eq}}$ for $\left\|u_{k_{eq}}\right\| \leq u_0$ yields motion in the sliding manifold $s = 0$. To proof convergence to this domain, consider the case $\left\|u_{k_{eq}}\right\| > u_0$, but in compliance with condition (2.31). From (2.29) to

(2.32) it follows that

$$s_{k+1} = \left(s_k + \left(CA^* - C\right)x_k + CD^* r_k\right)\left(1 - \frac{u_0}{\|u_{k_{eq}}\|}\right) \text{ with } u_0 < \|u_{k_{eq}}\|. \tag{2.33}$$

Thus,

$$\begin{aligned} \|s_{k+1}\| &= \|\left(s_k + \left(CA^* - C\right)x_k + CD^* r_k\right)\|\left(1 - \frac{u_0}{\|u_{k_{eq}}\|}\right) \\ &\leq \|s_k\| + \|\left(CA^* - C\right)x_k + CD^* r_k\| - \frac{u_0}{\|(CB^*)^{-1}\|} \\ &< \|s_k\|, \end{aligned} \tag{2.34}$$

due to (2.31).

Hence $\|s_k\|$ decreases monotonically and, after a finite number of steps, $\|u_{k_{eq}}\| < u_0$ is achieved. Discrete time sliding mode will take place from the next sampling point onwards.

Control (2.32) provides chatter-free motion in the manifold $s = 0$ as shown in Figure 2.3, in contrast to the direct implementation of discontinuous control in Figure 2.2, producing discretization chattering in the vicinity of the sliding manifold. Similar to the case of continuous time systems, the equation $s = Cx = 0$ enables the reduction of system order, and the desired system dynamics in *sliding mode* can be designed by appropriate choice of matrix C.

2.3 OPTIMAL CONTROL AND INVERSE OPTIMAL CONTROL

This section is devoted to briefly discuss optimal control and inverse optimal control.

Consider the nonlinear discrete time affine system

$$x(k+1) = f(x(k)) + g(x(k))u(k), \qquad x(0) = x_0, \tag{2.35}$$

with the associated meaningful cost functional

$$V(x(k)) = \sum_{n=k}^{\infty} l(x(n)) + u^T(n) R(x(n)) u(n), \tag{2.36}$$

where $x(k) \in \mathbb{R}^n$ is the state of the system at time $k \in \mathbb{Z}^+ \cup 0 = \{0,1,2,\ldots\}$; $u(k) \in \mathbb{R}^m$ is the control input, $f : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ and $g : \mathbb{R}^n \longrightarrow \mathbb{R}^{n\times m}$ are smooth mappings; $f(0) = 0$ and $g(x(k)) \neq 0$ for all $x(k) \neq 0$; $V : \mathbb{R}^n \longrightarrow \mathbb{R}_+$; $l : \mathbb{R}^n \longrightarrow \mathbb{R}_+$ is a positive semidefinite[1] function and $R : \mathbb{R}^n \longrightarrow \mathbb{R}^{m\times m}$ is a real symmetric positive definite[2] weighting matrix. The meaningful cost functional (2.36) is a performance measure [36]. The entries of R can be functions of the system state in order to vary the weighting on control effort according to the state value [36]. Considering the state feedback control design problem, we assume that the full state

$x(k)$ is available.

Equation (2.36) can be rewritten as

$$\begin{aligned} V(x(k)) &= l(x(k)) + u^T(k) R(x(k)) u(k) \\ &\quad + \sum_{n=k+1}^{\infty} (x(n)) + u^T(n) R(x(n)) u(n) \\ &= l(x(k)) + u^T(k) R(x(k)) u(k) + V(x(k+1)), \end{aligned} \tag{2.37}$$

where we require the boundary condition $V(0) = 0$ so that $V(x(k))$ becomes a Lyapunov function.

From Bellman's optimality principle [12, 43], it is known that, for the infinite horizon optimization case, the value function $V(x(k))$ becomes time invariant and satisfies the discrete time (DT) Bellman equation [5, 12, 57]

$$V(x(k)) = \min_{u(k)} \left\{ l(x(k)) + u^T(k) R(x(k)) u(k) + V(x(k+1)) \right\}, \tag{2.38}$$

where $V(x(k+1))$ depends on both $x(k)$ and $u(k)$ by means of $x(k+1)$ in (2.35). Note that the DT Bellman equation is solved backward in time [5].

To determine the optimal control law $u(k)$ in (2.38) is equivalent to calculate the gradient of (2.38) on the right-hand side with respect to $u(k)$; then

$$\begin{aligned} 0 &= 2R(x(k)) u(k) + \frac{\partial V(x(k+1))}{\partial u(k)} \\ &= 2R(x(k)) u(k) + g^T(x(k)) \frac{\partial V(x(k+1))}{\partial x(k+1)}. \end{aligned} \tag{2.39}$$

Therefore, the optimal control law is formulated as

$$u^*(k) = -\frac{1}{2} R^{-1}(x(k)) g^T(x(k)) \frac{\partial V(x(k+1))}{\partial x(k+1)}, \tag{2.40}$$

with the boundary condition $V(0) = 0$; $u^*(k)$ is used when we want to emphasize that the control law $u(k)$ is optimal.

Substituting (2.40) in (2.38) results in the following discrete time Hamilton-Jacobi-Bellman (HJB) equation:

$$\begin{aligned} &l(x(k)) + V(x(k+1)) - V(x(k)) + \frac{1}{4} \frac{\partial V^T(x(k+1))}{\partial x(k+1)} \times \\ &g(x(k)) R^{-1}(x(k)) g^T(x(k)) \frac{\partial V(x(k+1))}{\partial x(k+1)} = 0. \end{aligned} \tag{2.41}$$

Nevertheless, solving the partial differential equation (2.41) for $V(x(k))$ is not simple.

To solve the HJB constitutes the main drawback in order to obtain the optimal control problem solution for nonlinear systems [27, 75, 40]. To overcome this drawback, we propose discrete time *inverse optimal control* stated as follows.

Definition 2 (Inverse Optimal Control [58]) *Let the control law*

$$u^*(k) = -\frac{1}{2}R^{-1}(x(k))\, g^T(x(k))\, \frac{\partial V(x(k+1))}{\partial x(k+1)} \tag{2.42}$$

be inverse optimal (globally) stabilizing if:

(i) It achieves (global) asymptotic stability of $x = 0$ *for system* (2.35);

(ii) $V(x(k))$ *is (radially unbounded) positive definite function such that inequality*

$$\begin{aligned} \overline{V} &:= V(x(k+1)) - V(x(k)) + u^{*T}(k)\, R(x(k))\, u^*(k) \\ &\leq 0 \end{aligned} \tag{2.43}$$

is satisfied.

When we select $l(x(k)) := -\overline{V} \geq 0$, *then* $V(x(k))$ *is a solution for HJB equation* (2.41).

Due to the fact the inverse optimal control problem is based on the knowledge of $V(x(k))$ as established in Definition 2, we introduce the following definition.

Definition 3 (Control Lyapunov Function [8, 34]) *Let* $V(x(k))$ *be a radially unbounded function, with* $V(x(k)) > 0$, $\forall x(k) \neq 0$, *and* $V(0) = 0$. *If for any* $x(k) \in \mathbb{R}^n$, *there exist real values* $u(k)$ *such that*

$$\Delta V(x(k), u(k) < 0,$$

where the Lyapunov difference $\Delta V(x(k), u(k))$ *is defined as*

$$V(x(k+1)) - V(x(k)) = V(f(x(k)) + g(x(k))\, u(k)) - V(x(k)).$$

Then $V(\cdot)$ *is said to be a "discrete time control Lyapunov function" (CLF) for system (2.35).*

Hence, in order to solve the inverse optimal control we propose a CLF $V(x(k))$, such that (i) and (ii) are guaranteed. That is, instead of solving (2.41) for $V(x(k))$, we propose a control Lyapunov function $V(x(k))$ with the form

$$V(x(k)) = \frac{1}{2}x^T(k)\, Px(k), \quad P = P^T > 0 \tag{2.44}$$

for control law (2.42) in order to ensure stability of the equilibrium point $x(k) = 0$ of

system (2.35), which will be achieved by defining an appropriate matrix P. Moreover, it will be established that control law (2.42) with (2.44), which is referred to as the *inverse optimal* control law, optimizes a meaningful cost functional as given in (2.36).

Consequently, by considering $V(x(k))$ as in (2.44), control law (2.42) takes the following form:

$$\begin{aligned}\alpha(x(k)) &:= u^*(k)\\ &= -\tfrac{1}{2}\left(R(x(k)) + \tfrac{1}{2}g^T(x(k))P\times\right.\\ &\quad \left. g(x(k))\right)^{-1} g^T(x(k))\,P\,f(x(k)).\end{aligned} \tag{2.45}$$

It is worth pointing out that P and $R(x(k))$ are positive definite and symmetric matrices; hence, the existence of the inverse for (2.45) is ensured.

Theorem 2.1: [58, 59]

Consider the affine discrete time nonlinear system (2.35). If there exists a matrix $P = P^T > 0$ such that the following inequality holds:

$$\begin{aligned}V_f(x(k)) - \tfrac{1}{4}P_1^T(x(k))\left(R(x(k)) + P_2(x(k))\right)^{-1}\times\\ P_1(x(k)) \le -\zeta_Q\|x(k)\|^2,\end{aligned} \tag{2.46}$$

where $V_f(x(k)) = [V(f(x(k))) - V(x(k))]$, with $V(f(x(k))) = \frac{1}{2}f^T(x(k))\,P\,f(x(k))$ and $\zeta_Q > 0$; $P_1(x(k))$ and $P_2(x(k))$ as defined in (2.45); then, the equilibrium point $x(k) = 0$ of system (2.35) is globally exponentially stabilized by the control law (2.45), with CLF (2.44).

Moreover, with (2.44) as a CLF, this control law is inverse optimal in the sense that it minimizes the meaningful functional given by

$$V(x(k)) = \sum_{k=0}^{\infty}\left(l(x(k)) + u^T(k)\,R(x(k))\,u(k)\right) \tag{2.47}$$

with

$$l(x(k)) = -\overline{V}\big|_{u^*(k)=\alpha(x(k))}. \tag{2.48}$$

In [58] a fixed, symmetric, and positive definite matrix P for (2.45) is proposed in order to guarantee condition (2.46).

2.4 DISCRETE TIME HIGH-ORDER NEURAL NETWORKS

The use of multilayer neural networks is well known for pattern recognition and for modeling of static systems. The neural network (NN) is trained to learn an input-output

map. Theoretical works have proven that, even with just one hidden layer, a NN can uniformly approximate any continuous function over a compact domain, provided that the NN has a sufficient number of synaptic connections [20, 28, 33].

For control tasks, extensions of the first-order Hopfield model, called recurrent high-order neural networks (RHONN), which present more interactions among the neurons, are proposed in [53, 68]. Additionally, the RHONN model is very flexible and allows to incorporate to the neural model a priori information about the system structure. In this section, it is considered the identification problem of a nonlinear system formulated by

$$x(k+1) = F(x(k), u(k)), \tag{2.49}$$

where $x \in \Re^n$, $u \in \Re^m$, and $F \in \Re^n \times \Re^m \longrightarrow \Re^n$ is a nonlinear function. Then, consider the following discrete time RHONN:

$$\hat{x}_i(k+1) = w_i^T z_i(\hat{x}(k), \rho(k)), \; i = 1, \ldots, n, \tag{2.50}$$

where $\hat{x}_i \; (i = 1, 2, \cdots, n)$ is the state of the i th neuron, w_i (i = 1,2, …, n) is the respective on-line adapted weight vector, and $z_i(\hat{x}(k), \rho(k))$ is given by

$$z_i(\hat{x}(k), \rho(k)) = \begin{bmatrix} z_{i_1} \\ z_{i_2} \\ \vdots \\ z_{i_{M_i}} \end{bmatrix} = \begin{bmatrix} \prod_{j \in I_1} \xi_{ij}^{d_{ij}(1)} \\ \prod_{j \in I_2} \xi_{ij}^{d_{ij}(2)} \\ \vdots \\ \prod_{j \in I_{M_i}} \xi_{ij}^{d_{ij}(M_i)} \end{bmatrix}, \tag{2.51}$$

where M_i is the respective number of high-order connections, $\{I_1, I_2, \cdots, I_{M_i}\}$ is a collection of non-ordered subsets of $\{1,2, \cdots, n + m\}$, n is the state dimension, m is the number of external inputs, with d_{j_i} being nonnegative integers. As can be seen M_i is introduced to define $z_i(\hat{x}(k), \rho(k))$, which is used on (2.50), and ξ_i is defined as follows

$$\xi_i = \begin{bmatrix} \xi_{i_1} \\ \vdots \\ \xi_{i_n} \\ \xi_{i_{n+1}} \\ \vdots \\ \xi_{i_{n+m}} \end{bmatrix} = \begin{bmatrix} S(\hat{x}_1) \\ \vdots \\ S(\hat{x}_n) \\ \rho_1 \\ \vdots \\ \rho_m \end{bmatrix}. \tag{2.52}$$

In (2.52), $\rho = [\rho_1, \rho_2, \ldots, \rho_m]^T$ is the input vector to the neural network, and $S(\bullet)$ is defined by

$$S(\varsigma) = \frac{1}{1+\exp(-\beta\varsigma)}, \beta > 0, \tag{2.53}$$

where ς is any real value variable.

Consider the problem to approximate the general discrete time nonlinear system (2.49), by the following discrete time RHONN series-parallel representation [72, 73]:

$$x_i(k+1) = w_i^{*T} z_i(x(k)) + \rho(k)) + \varepsilon_{z_i}, i = 1, \cdots, n, \tag{2.54}$$

where x_i is the i th plant state, and ε_{z_i} is a bounded approximation error, which can be reduced by increasing the number of the adjustable weights [32, 72]. Assume that there exists an ideal weight vector $\boldsymbol{w_i^*}$ such that $\|\boldsymbol{\varepsilon_{z_i}}\|$ can be minimized on a compact set $\Omega_{z_i} \subset \Re^{L_i}$. The ideal weight vector $\boldsymbol{w_i^*}$ is an artificial quantity required for analytical purpose [32, 72]. In general, it is assumed that this vector exists and is constant but unknown. Let us define its estimate as w_i and the estimation error as

$$\tilde{w}_i(k) = w_i(k) - w_i^*. \tag{2.55}$$

The estimate w_i is used for stability, which will be discussed later. Since $\boldsymbol{w_i^*}$ is constant, then $\tilde{w}_i(k+1) - \tilde{w}_i(k) = w_i(k+1) - w_i(k), \forall k \in 0 \cup \mathbb{Z}^+$. From (2.50) three possible models can be derived:

Parallel model

$$\hat{x}_i(k+1) = w_i^T z_i(\hat{x}(k), \rho(k)), i = 1, \ldots, n, \tag{2.56}$$

Series-Parallel model

$$\hat{x}_i(k+1) = w_i^T z_i(x(k), \rho(k)), i = 1, \ldots, n, \tag{2.57}$$

Feedforward model (HONN)

$$\hat{x}_i(k) = w_i^T z_i(\rho(k)), i = 1, \ldots, n, \tag{2.58}$$

where $\hat{x}$ is the NN state vector, x is the plant state vector, and ρ is the input vector to the NN.

2.5 EKF TRAINING ALGORITHM

The well-known training approach for recurrent neural networks (RNN) is the back propagation through time learning [84]. However, it is a first-order gradient descent method and hence its learning speed could be very slow [42]. Recently, extended Kalman filter (EKF)-based algorithm has been introduced to train neural networks [7, 25, 72].

With the EKF-based algorithm, the learning convergence is improved [42]. The EKF training of neural networks, both feedforward and recurrent ones, has proven to be reliable and practical for many applications [25].

The Kalman filtering (KF) estimates the state of a linear system with additive state and output white noises [31, 76]. For KF-based neural network training, the network weights become the states to be estimated. In this case, the error between the neural network output and the measured plant output can be considered as additive white noise. Due to the fact that the neural network mapping is nonlinear, an EKF type is required (see [68] and references therein).

The training goal is to determine the optimal weight values, which minimize the prediction error. The EKF-based training algorithm is described by [31, 72]:

$$\begin{aligned} K_i(k) &= P_i(k) H_i(k) \left[R_i(k) + H_i^T(k) P_i(k) H_i(k)\right]^{-1} \\ w_i(k+1) &= w_i(k) + \eta_i K_i(k) \left[y(k) - \hat{y}(k)\right] \\ P_i(k+1) &= P_i(k) - K_i(k) H_i(k)^T P_i(k) + Q_i(k), \end{aligned} \tag{2.59}$$

where $P_i \in \Re^{L_i \times L_i}$ is the estimation error associated covariance matrix, $w_i \in \Re^{L_i}$ is the weight (state) vector, L_i is the total number of neural network weights, $y \in \Re^m$ is the measured output vector, $\hat{\boldsymbol{y}} \in \Re^m$ is the network output, η_i is a design parameter, $K_i \in \Re^{L_i \times m}$ is the Kalman gain matrix, $Q_i \in \Re^{L_i \times L_i}$ is the state noise associated covariance matrix, $R_i \in \Re^{m \times m}$ is the measurement noise associated covariance matrix, and $H_i \in \Re^{L_i \times m}$ is a matrix, for which each entry (H_{ij}) is the derivative of one of the neural network output, $(\hat{\boldsymbol{y}})$, with respect to one neural network weight, (w_{ij}), as follows:

$$H_{ij}(k) = \left[\frac{\partial \hat{y}(k)}{\partial w_{ij}(k)}\right]_{w_i(k)=\hat{w}_i(k+1)}, \tag{2.60}$$

where $i = 1,\ldots, n$ and $j = 1,\ldots,L_i$.

Usually P_i, Q_i, and R_i are initialized as diagonal matrices, with entries $P_i(0)$, $Q_i(0)$, and $R_i(0)$, respectively. It is important to note that $H_i(k)$, $K_i(k)$, and $P_i(k)$ for the EKF are bounded [76]. Therefore, there exist constants $\overline{\boldsymbol{H}}_i > 0, \overline{\boldsymbol{K}}_i > 0$, and $\overline{\boldsymbol{P}}_i > 0$ such that

$$\begin{aligned} \|H_i(k)\| &\leq \overline{H}_i \\ \|K_i(k)\| &\leq \overline{K}_i \\ \|P_i(k)\| &\leq \overline{P}_i. \end{aligned} \tag{2.61}$$

The convergence analysis for the RHONN trained by EFK is detailed in [6, 72]. The measurement and process noises are typically characterized as zero-mean, white noises with covariances given by $\delta_{k,j} R_i(k)$ and $\delta_{k,j} Q_i(k)$, respectively, where $\delta_{k,j}$ is a Kronecker delta function (0 for $k \neq l$ and 1 for $k = l$) [32]. In order to simplify the notation in this

dissertation the covariances will be represented by their respective associated matrices, $R_i(k)$ and $Q_i(k)$ for the noises and $P_i(k)$ for the prediction error.

2.6 NEURAL CONTROL

To control a system is to force it to behave in a desired way. How to express this "desired behavior"depends primarily on the task to be solved; however, the dynamics of the system, the actuators, the measurement equipment, the available computational power, etc., influence the formulation of the desired behavior as well. Although the desired behavior obviously is very dependent on the application, the need to express it in mathematical terms suited for practical design of control systems seriously limits the means of expressing this behavior. At the higher level, it is customary to distinguish two basic types of problems [56]:

Regulation problem. The fundamental desired behavior is to keep the output of the system at a constant level regardless of the disturbances acting on the system.

Tracking problem. The fundamental desired behavior is to force the system output to track a reference trajectory closely.

Neural networks have become a well-established methodology as exemplified by their applications to identification and control of general nonlinear and complex systems [29]; the use of high-order neural networks for modeling and learning has recently increased [74]. Specifically, the problem of designing robust neural controllers for nonlinear systems with uncertainties and disturbances, which guarantees stability and trajectory tracking, has received increasing attention lately.

Using neural networks, control algorithms can be developed to be robust to uncertainties and modeling errors. The most used NN structures are: *feedforward* networks and *recurrent* ones [74]. The last type offers a better suited tool to model and control nonlinear systems [64].

The neural control problem can be approached in two different ways:

Direct control system design. Direct means that the controller is a neural network. A neural network controller is often advantageous when the real time platform available prohibits complicated solutions. The implementation is simple while the design and tuning are difficult.

Indirect control system design. This class of designs is always model-based. The idea is to use a neural network to model the system to be controlled; this model is then employed in a more conventional controller design. The model is typically trained in advance, but the controller is designed on line. As will appear, the indirect design is very flexible; thus it is the most appropriated for most of the common control problems.

The increasing use of NN to modeling and control is in great part due to the following features that make them particularly attractive [26]:

NN are universal approximators: It has been proven that any continuous nonlinear

function can be approximated arbitrarily well over a compact set by a multilayer neural network which consists of one or more hidden layers [19]. Learning and adaptation: The intelligence of neural networks comes from their generalization ability with respect to unknown data. On-line adaptation of the weights is possible.

Multivariable systems: NN have many inputs and outputs, which makes it easy to model multivariable systems.

2.7 PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) was developed in [35] as a stochastic optimization algorithm based on social simulation models. The algorithm employs a population of search points, which moves stochastically in the search space.

The population is called the swarm and its individuals are called the particles; a notation retained by nomenclature used for similar models in social sciences and particle physics. The swarm is defined as a set:

$$Sw = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}, \tag{2.62}$$

of N particles (candidate solutions), defined as

$$\mathbf{x}_i = (\mathbf{x}_{i1}, \mathbf{x}_{i2}, \ldots, \mathbf{x}_{in})^T \in A, i = 1, 2, \ldots, N. \tag{2.63}$$

The objective function, $f(\mathrm{x})$, is assumed to be available for all points in A. Thus, each particle has a unique function value, $f_i = f(\mathrm{x}_i) \in Y$. The particles are assumed to move within the search space, A, iteratively, which is possible by adjusting their position using a proper position shift, called velocity, and denoted as

$$v_i = (v_{i1}, v_{i2}, \ldots, v_{in})^T, i = 1, 2, \ldots, N. \tag{2.64}$$

Velocity is also adapted iteratively to render particles capable of potentially visiting any region of A. If k denotes the iteration counter, then the current position of the k th particle and its velocity will be henceforth denoted as $\mathrm{x}_i(k)$ and $v_i(k)$, respectively. Velocity is updated based on information obtained in previous steps of the algorithm, which is implemented in terms of a memory, where each particle can store the best position it has ever visited during its search. For this purpose, besides the swarm, S, which contains the current positions of the particles, PSO maintains also a memory set:

$$P = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N\}, \tag{2.65}$$

which contains the best positions,

$$\mathbf{p}_i = (\mathbf{p}_{i1}, \mathbf{p}_{i2}, \ldots, \mathbf{p}_{in})^T \in A, i = 1, 2, \ldots, N, \tag{2.66}$$

ever visited by each particle. These positions are defined as

$$\mathbf{p}_i(k) = argmin_k f_i(k), \tag{2.67}$$

where k stands for the iteration counter. The algorithm approximates the global minimizer with the best position ever visited by all particles. Let g be the index of the best position with the lowest function value in P at a given iteration k,

$$\mathbf{p}_g(k) = argmin_k f(\mathbf{p}_i(k)). \tag{2.68}$$

Then, the PSO is defined by the following equations [22, 23, 35] :

$$v_{ij}(k+1) = v_{ij}(k) + c_1 \mathrm{R}_1 \left(\mathbf{p}_{ij}(k) - \mathbf{x}_{ij}(k)\right) + c_2 \mathrm{R}_2 \left(\mathbf{p}_{gj}(k) - \mathbf{x}_{ij}(k)\right), \tag{2.69}$$

$$\begin{gathered} x_{ij}(k+1) = \mathbf{x}_{ij}(k) + v_{ij}(k+1), \\ i = 1, 2, \ldots, N, j = 1, 2, \ldots, n, \end{gathered} \tag{2.70}$$

where k denotes the iteration counter; R_1 and R_2 are random variables uniformly distributed within [0,1]; and c_1, c_2 are weighting factors, also called the cognitive and social parameter, respectively [60].

The last two terms in (2.69) enable each particle to perform a local search around its individual best position p_{ij} and the swarm best position p_{gj}. The first term in (2.69) enables each particle to perform a global search by exploring a new search space. The multi-agent (particles) searching and information-sharing mechanism in PSO enables a fast and efficient search for the optimal solution. In many cases, the PSO algorithm yields superior performance to other evolutionary computation algorithms, such as genetic algorithms.

NOTES

[1] A function $l(z)$ is positive semidefinite (or nonnegative definite) function if for all vectors z, $l(z) \geq 0$. In other words, there are some vectors z for which $l(z) = 0$, and for all others z, $l(z) > 0$ [36].

[2] A real symmetric matrix R is positive definite if $z^T R z > 0$ for all $z \neq 0$ [36].

3 Modeling of Wind Turbines

In this chapter, an overview of wind turbines and electric generators used for wind energy is presented. Each of the generators have different schemes of operation and this chapter is intended to explore in a general way each of them. Having analyzed each generation scheme in section 3.2, the mathematical models for the doubly fed induction generator (DFIG) and the DC Link between the rotor side converter (RSC) and the grid side converter (GSC) are presented. For details on obtaining the mathematical models presented in this section, we refer to Appendix B.

3.1 WIND ENERGY GENERATION SYSTEMS

Wind energy technology has evolved rapidly over the last three decades with increasing rotor diameters and the use of sophisticated power electronics to allow operation at variable rotor speed.

The DFIG-based wind turbine has been used widely when connecting to the AC grid. As a variable-speed wind generator, the DFIG has a number of advantages compared to fixed-speed generators, including that the machine can operate in sub-synchronous mode, synchronous mode, as well as super-synchronous mode, decoupled control of active and reactive power, better energy capture, mechanical stress reduction, and low cost with the development of the power electronics converters [50]. The conventional topology, the bi-directional buck-boost DC-DC converter, is connected between the fixed DC Link voltage. The RSC is connected to the rotor controlling the active and reactive power of the DFIG stator. The GSC regulates the DC Link voltage with the AC side to the converter connected to the grid [61, 65].

A typically simplified configuration of a DFIG wind turbine is shown schematically in Figure 3.1; a wound-rotor induction generator with slip rings is used to take current into or out of the rotor winding and variable-speed operation is obtained injecting a controllable voltage into the rotor at slip frequency. The rotor winding is fed through a variable-frequency power converter, typically based on two AC/DC insulated-gate bipolar transistor (IGBT)-based voltage source converters, namely, RSC and GSC, linked by a DC Link. This configuration in known as variable frequency AC/DC/AC converter (VFC), which only needs to handle a fraction (25%-30%) of the total DFIG power to achieve full control of the generator. The GSC is connected to the stator terminals directly or through a step-up transformer. Both the RSC and GSC are four-quadrant converters which allow bi-directional power flow; different techniques have been proposed for this configuration control. In [18], a complete overview of control systems for the DFIG based on different configurations is presented, and on that paper the importance of this type of generator for wind energy applications is highlighted. In [46], a steady-state stability analysis for the effect of rotor excitation voltage of the DFIG and experimental results are given to

demonstrate the effectiveness of the theoretical analysis.

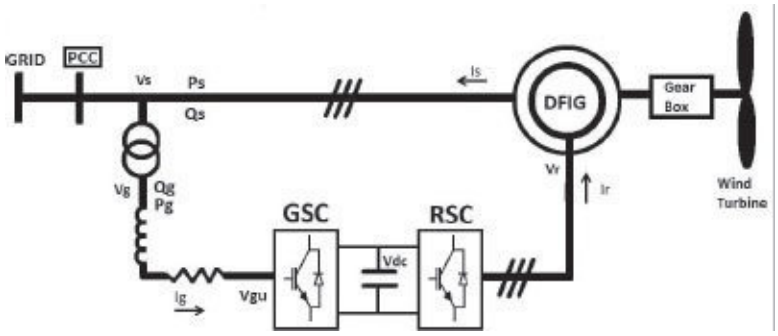

Figure 3.1 Variable-speed DFIG with IGBT converters.

3.1.1 WIND TURBINES

Wind turbines produce electricity by using the power of the wind to drive an electrical generator. Wind passes over the blades, generating lift and exerting a turning force. The rotating blades turn a shaft inside the nacelle, which goes into a turning force. The rotating blades increase the rotational speed to the appropriate value for the generator, which uses magnetic fields to convert the rotational energy into electrical energy. The power output goes to a transformer, which converts the electricity from the generator at around 700 V to the appropriate voltage for the power collection system, typically 33kV [9].

A wind turbine extracts kinetic energy from the swept area of the blades. The power in the airflow is given by

$$P_{air} = \frac{1}{2}\rho A v^3, \tag{3.1}$$

where ρ is the air density (approximately 1.225 kg/m^3), A is the swept area of rotor (m^2), and v is upwind free wind speed (m/s) [9].

Although (3.1) gives the power available in the wind, the power transferred to the wind turbine rotor is reduced by the power coefficient, C_p:

$$C_p = \frac{P_{wind\ turbine}}{P_{air}}, \tag{3.2}$$

$$P_{wind\ turbine} = C_p P_{air} = \frac{1}{2} C_p \rho A v^3. \tag{3.3}$$

A maximum value of C_p is defined by the Betz limit, which states that a turbine can never extract more than 59.3% of the power from an air stream. In reality, wind turbine rotors have maximum C_p values in the range 25-45%. It is also conventional to define a tip-speed ratio, λ, as

$$\lambda = \frac{\omega R}{v}, \tag{3.4}$$

where ω is the rotational speed of rotor, R is the radius to tip of rotor, and v is the upwind free wind speed [9].

The tip ratio λ and the power coefficient C_p are dimensionless and so can be used to describe the performance of any size of wind turbine rotor. Figure 3.2 shows that the maximum power coefficient is only achieved at a single tip-speed ratio and for a fixed rotational speed of the wind turbine it only occurs at a single wind speed. Hence, one argument for operating a wind turbine at variable rotational speed is that it is possible to operate at maximum C_p over a range of wind speeds [9].

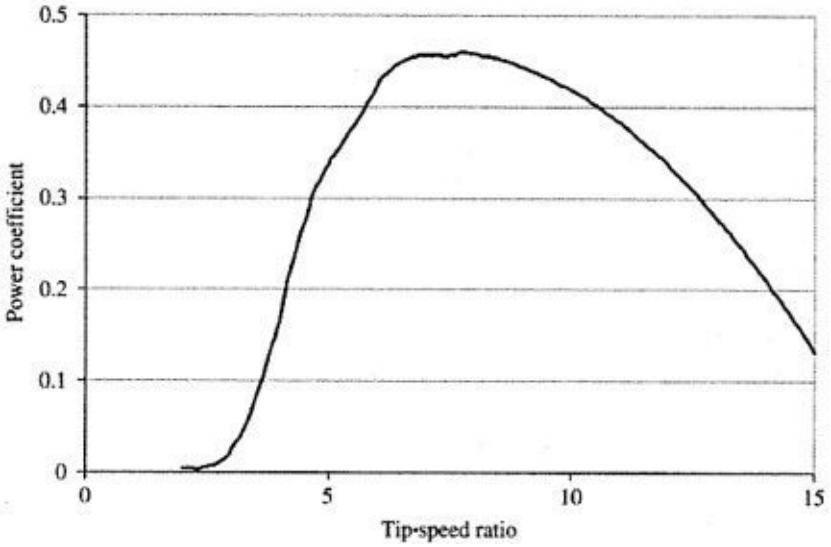

Figure 3.2 Illustration of power coefficient/tip-speed ratio curve, C_p/λ.

The power output of a wind turbine at various wind speeds is conventionally described by its power curve. The power curve gives the steady-state electrical power output as a function of the wind speed at the hub height and is generally measured using 10 *min* average data. An example of power curve is given in Figure 3.3 [9].

The power curve has three key points on the velocity scale:

1. Cut-in wind speed: the minimum wind speed at which the machine will deliver useful power
2. Rated wind speed: the wind speed at which rated power is obtained (rated power is generally the maximum power output of the electrical generator)
3. Cut-out wind speed: the maximum wind speed at which the turbine is allowed to deliver power (usually limited by engineering loads and safety constrains)

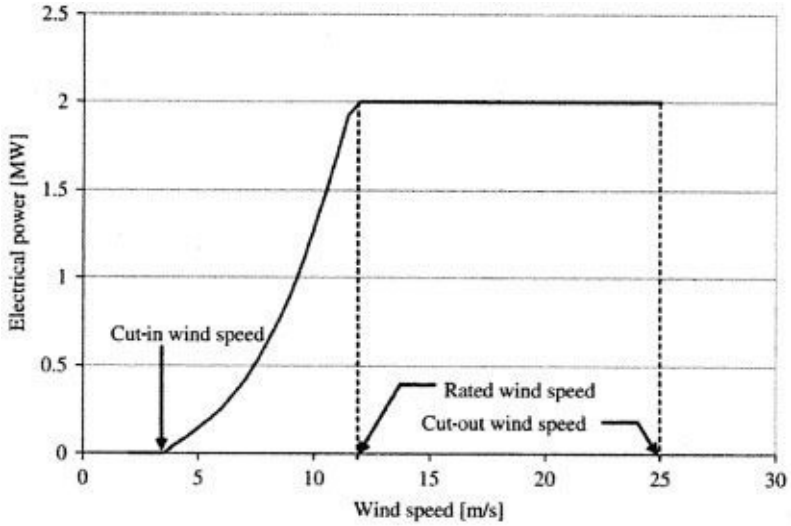

Figure 3.3 Power curve for a 2 MW wind turbine.

Below the cut-in speed, of about 5 *m/s*, the wind turbine remains shut down as the speed of the wind is too low for useful energy production. Then once in operation, the power output increases following a broadly cubic relationship with wind speed until rated wind speed is reached. Above rated wind speed the aerodynamic rotor is arranged to limit the mechanical power extracted from the wind and so reduce the mechanical loads on the drive train. Then at very high wind speeds the turbine is shut down [9].

The choice of cut-in, rated, and cut-out wind speed is made by the wind turbine designer who, for typical wind conditions, will try to balance obtaining maximum energy extraction with controlling the mechanical loads of the turbine. For a mean annual site wind speed V_m of 8 *m/s* typical values will be approximately

Cut-in wind speed: 5 *m/s*, 0.6 V_m,

Rated wind speed: 12-14*m/s*, 1.5-1.75 V_m,

Cut-out wind speed: 25 *m/s*, 3 V_m.

Power curves for existing machines can normally be obtained from the turbine manufacturer. They are found by field measurements, where an anemometer is placed on a mast reasonably close to the wind turbine, not on the turbine itself or too close to it, since the turbine may create turbulence, making wind speed measurements unreliable [9].

3.1.2 WIND TURBINE ARCHITECTURES

There are a large number of architecture choices available to the designer of a wind turbine and, over the years, most of these have been explored. However, commercial designs for electricity generation have now converged to horizontal axis, three-bladed, upwind turbines. The largest machines tend to operate at variable speeds, whereas smaller, simpler turbines are of fixed speed [9].

Modern electricity-generating wind turbines now use three-bladed upwind rotors, although two-bladed, and even one-bladed, rotors were used in earlier commercial turbines. Reducing the number of blades means that the rotor has to operate at a higher

rotational speed in order to extract the wind energy passing through the rotor disk. Although a high rotor speed is attractive in that it reduces the gearbox ratio required, a high blade tip speed leads to increased aerodynamic noise and increased blade drag losses. Most importantly, three-bladed rotors are visually more pleasing than other designs and so these are now always used on large electricity-generating turbines [9].

3.1.2.1 Fixed-Speed Wind Turbines

Fixed-speed wind turbines are electrically fairly simple devices consisting of an aerodynamic rotor driving a low-speed shaft, a gearbox, a high-speed shaft, and an induction generator. From the electrical system viewpoint they are perhaps best considered as large fan drives with torque applied to the low-speed shaft from the wind flow [9].

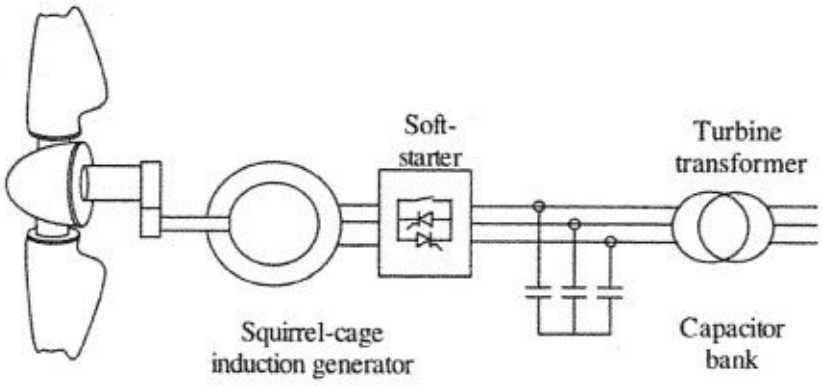

Figure 3.4 Schematic of a fixed-speed wind turbine.

Figure 3.4 illustrates the configuration of a fixed-speed wind turbine. It consists of a squirrel-cage induction generator coupled to the power system through a turbine transformer. The generator operating slip changes slightly as the operating power level changes and the rotational speed is therefore not entirely constant. However, because the operating slip variation is generally less than 1%, this type of wind generation is normally referred to as fixed speed [9].

Squirrel-cage induction machines consume reactive power and so it is conventional to provide power factor correction capacitors at each wind turbine. The function of the soft-starter unit is to build up the magnetic flux slowly, minimizing transient currents during energization of the generator. Also, by applying the network voltage slowly to the generator, once energized, it brings the drive train slowly to its operating rotational speed [9].

3.1.2.2 Variable-Speed Wind Turbines

As the size of wind turbines has become larger, the technology has switched from fixed speed to variable speed. The drivers behind these developments are mainly the ability to comply with Grid Code connection requirements and the reduction in mechanical loads achieved with variable-speed operation. Currently the most common variable-speed wind turbine configurations are as follows [9]:

DFIG wind turbine, and

Fully rated converter (FRC) wind turbine based on a synchronous or induction generator.

Doubly Fed Induction Generator (DFIG) Wind Turbine

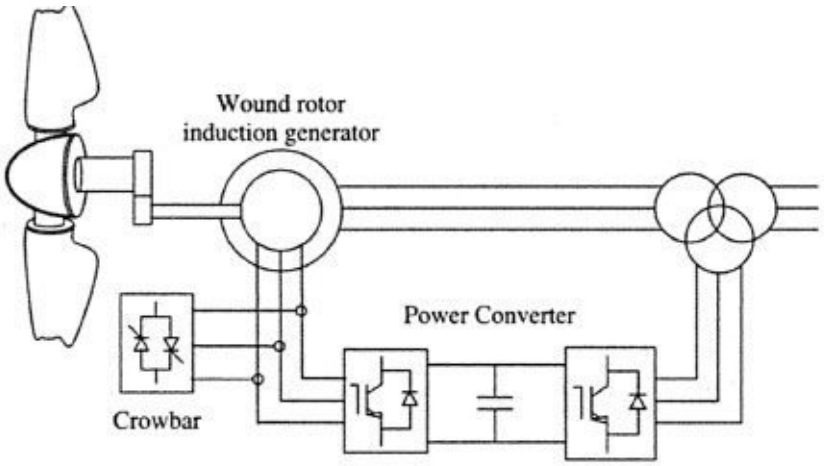

Figure 3.5 Typical configuration of a DFIG wind turbine.

A typical configuration of a DFIG wind turbine is shown schematically in Figure 3.5. It uses a wound-rotor induction generator with slip rings to take current into or out of the rotor winding and variable-speed operation is obtained injecting a controllable voltage into the rotor at slip frequency. The rotor winding is fed through a variable-frequency power converter, typically based on two AC/DC IGBT-based voltage source converters (VSCs), linked by a DC Link. The power converter decouples the network electrical frequency from the rotor mechanical frequency, enabling variable-speed operation of the wind turbine. The generator and converters are protected by voltage limits and an over-current crowbar [9].

A DFIG system can deliver power to the grid through the stator and rotor, while the rotor can also absorb power, which depends on the rotational speed of the generator. If the generator operates above synchronous speed, power will be delivered from the rotor through the converters to the network, and if the generator operates below synchronous speed, then the rotor will absorb power from the network through the converters [9].

FRC Wind Turbine

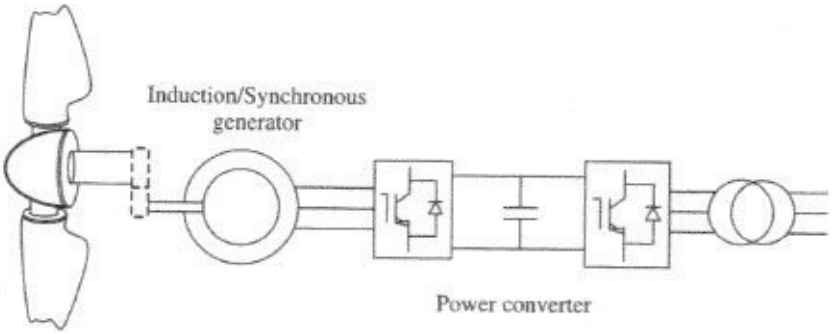

Figure 3.6 Typical configuration of an FRC-connected wind turbine.

The typical configuration of an FRC wind turbine is shown in Figure 3.6. This type of turbine may or may not include a gearbox and a wide range of electrical generator types

can be employed, for example, induction, wound rotor synchronous, or permanent magnet synchronous. As all of the power from the turbine goes through the power converters, the dynamic operation of the electrical generator is effectively isolated from the power grid. The electrical frequency of the generator may vary as the wind speed changes, while the grid frequency remains unchanged, allowing variable-speed operation of the wind turbine [9].

The power converters can be arranged in various ways. Whereas the generator side converter can be a diode rectifier or a pulse width modulation (PWM) voltage source converter, the network side converter is typically a PWM VSC. The strategy to control the operation of the generator and the power flows to the network depends very much on the type of power converter arrangement employed. The network-side converter can be arranged to maintain the DC bus voltage constant with torque applied to the generator controlled from the generator-side converter. Alternatively, the control philosophy can be reversed. Active power is transmitted through the converters with very little energy stored in the DC Link capacitor. Hence the torque applied to the generator can be controlled by the network-side converter. Each converter is able to generate or absorb reactive power independently [9].

3.2 DISCRETE TIME MATHEMATICAL MODELS

3.2.1 DOUBLY FED INDUCTION GENERATOR

The DFIG rotor and stator three-phase windings are assumed to be symmetrical and balanced, and the rotor speed can be below or above the synchronous speed of the stator flux [17]. The discrete time equations for the DFIG are developed in *d-q* axis per unit as in [69]:

$$\omega_r(k+1) = \omega_r(k) + \frac{t_s}{2H}(\tau_m(k) - \tau_e(k)), \tag{3.5}$$

$$i_s(k+1) = i_s(k) + t_s(A_{11}(k)\, i_s(k) + A_{12}(k)\, i_r(k)) + t_s(D_1 v_s(k) + B_1 u(k)), \tag{3.6}$$

$$i_r(k+1) = i_r(k) + t_s(A_{21}(k)\, i_s(k) + A_{22}(k)\, i_r(k)) + t_s(D_2 v_s(k) + B_2 u(k)), \tag{3.7}$$

$$\tau_e(k) = i_r(k)^T M_{\tau_e i_s}(k), \tag{3.8}$$

where

$$i_s(k)=\begin{bmatrix} i_{ds}(k) \\ i_{qs}(k) \end{bmatrix}, i_r(k)=\begin{bmatrix} i_{dr}(k) \\ i_{qr}(k) \end{bmatrix}, v_s(k)=\begin{bmatrix} v_{ds}(k) \\ v_{qs}(k) \end{bmatrix}, u(k) = \begin{bmatrix} v_{dr}(k) \\ v_{qr}(k) \end{bmatrix}$$

$$A_{11}(k)=\begin{bmatrix} -\frac{\omega_b r_s}{X_s \sigma} & \omega_b\left(1-\frac{\sigma-1}{\sigma}\omega_r(k)\right) \\ -\omega_b\left(1-\frac{\sigma-1}{\sigma}\omega_r(k)\right) & -\frac{\omega_b r_s}{X_s \sigma} \end{bmatrix},$$

$$A_{12}(k)=\begin{bmatrix} -\frac{\omega_b X_m r_r}{X_s X_r \sigma} & -\frac{\omega_b X_m}{X_s \sigma}\omega_r(k) \\ \frac{\omega_b X_m}{X_s \sigma}\omega_r(k) & -\frac{\omega_b X_m r_r}{X_s X_r \sigma} \end{bmatrix}$$

$$A_{21}(k)=\begin{bmatrix} -\frac{\omega_b X_m r_s}{X_s X_r \sigma} & \frac{\omega_b X_m}{X_r \sigma}\omega_r(k) \\ -\frac{\omega_b X_m}{X_r \sigma}\omega_r(k) & -\frac{\omega_b X_m r_s}{X_s X_r \sigma} \end{bmatrix},$$

$$A_{22}(k)=\begin{bmatrix} -\frac{w_b r_r}{X_r \sigma} & \omega_b\left(1-\frac{1}{\sigma}\omega_r(k)\right) \\ -\omega_b\left(1-\frac{1}{\sigma}\omega_r(k)\right) & -\frac{\omega_b r_r}{X_r \sigma} \end{bmatrix}$$

$$B_1=\begin{bmatrix} \frac{\omega_b X_m}{X_s X_r \sigma} & 0 \\ 0 & \frac{\omega_b X_m}{X_s X_r \sigma} \end{bmatrix}, D_1=\begin{bmatrix} -\frac{\omega_b}{X_s \sigma} & 0 \\ 0 & -\frac{\omega_b}{X_s \sigma} \end{bmatrix},$$

$$D_2=\begin{bmatrix} -\frac{\omega_b X_m}{X_s X_r \sigma} & 0 \\ 0 & -\frac{\omega_b X_m}{X_s X_r \sigma} \end{bmatrix}, B_2=\begin{bmatrix} \frac{\omega_b}{X_r \sigma} & 0 \\ 0 & \frac{\omega_b}{X_r \sigma} \end{bmatrix},$$

$$\sigma=1-\frac{X_m^2}{X_s X_r}, M_{\tau_e}=X_m\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

and ω_b is the synchrony frequency (*pu*), ω_r is the rotor speed (*pu*), i_{ds}, i_{qs}, i_{dr}, i_{qr} are the stator and rotor currents in the *d* and *q* axes, respectively (*pu*), v_{ds}, v_{qs}, v_{dr}, v_{qr} are the stator and rotor voltages in the *d* and *q* axes, respectively (*pu*), τ_e is the electromagnetic torque (*pu*), τ_m is the mechanical torque (*pu*), X_s is the stator selfreactance per phase (*pu*), X_r is the rotor self-reactance per phase (*pu*), X_m is the magnetization reactance (*pu*), r_s is the stator resistance per phase (*pu*), r_r is the rotor resistance per phase (*pu*), H is the moment of inertia (*sec*), and t_s is the sampling time. This mathematical model is developed in detail in Appendix B.1.

The DFIG parameters used for subsequent simulations were provided for the manufacturer, as appear in Table 3.1.

Table 3.1
Doubly Fed Induction Generator and DC Link Parameters

Symbol	Parameter	Value

X_m	Magnetizing Reactance	2.3175 *pu*
X_s	Stator Reactance	2.4308 *pu*
X_r	Rotor Reactance	2.4308 *pu*
r_s	Stator Windings Resistance	0.1609 *pu*
r_r	Rotor Windings Resistance	0.0502 *pu*
H	Angular Moment of Inertia	0.23 *sec*
ω_b	Base Angular Frequency	376.99112 *rad/sec*
P_b	Base Power	185.4*VA*
V_b	Base Voltage	179.63*V*
X_l	Three Phase Lines Reactance	0.0045 *pu*
r_g	Three Phase Lines Resistance	0.0014 *pu*
C	DC Link Capacitance	0.1854 *pu*

3.2.2 DC LINK

The GSC is connected to the grid source through a step-up transformer as shown in Figure 3.7. The GSC works like a rectifier to keep the DC voltage constant in the DC Link and to allow bi-directional power flow. The discrete time equations in the *d-q* axis for the DC Link are defined as

$$v_{dc}(k+1) = v_{dc}(k) + t_s\left(\frac{1}{Cv_{dc}(k)} v_{gs}^T(k) M_p i_g(k)\right), \tag{3.9}$$

$$i_g(k+1) = i_g(k) + t_s\left(A_g i_g(k) + B_g v_{gs}(k) - B_g u_g(k)\right), \tag{3.10}$$

with $v_{dc}(0) \neq 0$, where

$$A_g = \begin{bmatrix} -\frac{\omega_b R_g}{X_l} & \omega_s \\ -\omega_s & -\frac{\omega_b R_g}{X_l} \end{bmatrix}, B_g = \begin{bmatrix} \frac{\omega_b}{X_l} & 0 \\ 0 & \frac{\omega_b}{X_l} \end{bmatrix},$$

$$M_P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$i_g(k) = \begin{bmatrix} i_{dg}(k) \\ i_{qg}(k) \end{bmatrix}, v_{gs}(k) = \begin{bmatrix} v_{dgs}(k) \\ v_{qgs}(k) \end{bmatrix}, u_g(k) = \begin{bmatrix} v_{dg}(k) \\ v_{qg}(k) \end{bmatrix},$$

and v_{dc} is the DC Link voltage (*pu*), ω_b is the synchrony frequency (*pu*), i_{dg}, i_{qg} are the currents in the d and q axes, respectively (*pu*), v_{dgs}, v_{qgs}, v_{dg}, v_{qg} are the step-up

transformer and GSC voltages in the d and q axes, respectively (pu), R_g is the resistance of the three phase lines a, b, c (pu), X_l is the reactance of three phase lines a, b, c (pu), C is the capacitance of the DC Link (pu), and t_s is the sampling time. The DC Link parameters appear in Table 3.1. This mathematical model is developed in detail in Appendix B.2.

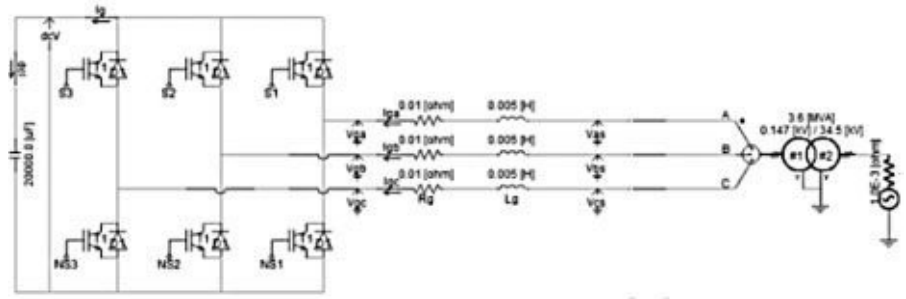

Figure 3.7 GSC connected to electrical grid by a step-up transformer.

As mentioned in Chapter 3, the doubly fed induction generator (DFIG) generation scheme is composed of the rotor side coverter (RSC) and the grid side converter (GSC). The rotor winding of the DFIG is connected directly to the RSC, which is used to control the electric torque (τ_e) and the stator reactive power (Q_s) by means of the rotor voltages. In general, the GSC is used to control the DC voltage (v_{dc}) of the capacitor in the DC Link and the reactive power (Q_g) or the three-phase voltage (v_g) in the step-up transformer, too. Hence, this generation configuration needs two controllers (DFIG and DC Link controllers) working together. The DFIG controller is implemented in the RSC and the DC Link controller is implemented in the GSC.

In this chapter different control algorithm applications for the DFIG scheme are presented. In Section 4.1, a block control sliding modes controller is developed for the RSC and GSC. Simulation results are presented to illustrate the performance of these controllers. In Section 4.2, the inverse optimal control is applied to the DFIG and GSC, and simulation results are presented to validate the performance of this algorithm.

4.1 BLOCK CONTROL SLIDING MODES

4.1.1 DFIG CONTROLLER

The variables to be controlled are the DFIG electric torque ($\tau_e(k)$) and the stator reactive power ($Q_s(k)$). The goals are: a) to track a time-varying electric torque trajectory ($\tau_e^{ref}(k)$), and b) to keep constant the electric power factor ($f_{ps1}(k)$) at the stator terminals through the stator reactive power ($Q_s(k)$) control. The electric torque ($\tau_e(k)$) and the stator reactive power ($Q_s(k)$) are defined, respectively, as

$$\tau_e(k) = i_r(k)^T M_{\tau_e} i_s(k), \tag{4.1}$$

and

$$Q_s(k) = v_s(k)^T M_Q i_s(k), \tag{4.2}$$

where $M_{\tau_e} = X_m \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, M_Q = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$.

As explained in [39, 77], an induction machine converts electrical power to mechanical power, $P_{electric} = P_{mechanic} + P_{losses}$. Considering P_{losses} negligible, the electrical power is fully converted to mechanical power $P_{electric} = P_{mechanic}$. The change of mechanical energy

in a rotational system is expressed as $P_{mechanic} = \frac{2}{n_{poles}}\omega_r \tau_e$. In steady-state, we can consider $\omega_r \approx \omega_{sync}$, then $P_{electric} \approx \frac{2}{n_{poles}}\omega_{sync}\tau_e$. Applying the *p.u.* transformation, the electric power can be rewritten as: $P \approx \tau_e$ [39]. From here we can see that controlled electric torque (τ_e) of the generator is equivalent to control of the active power thereof.

The electric torque reference $\left(\tau_e^{ref}(k)\right)$ is defined as

$$\tau_e^{ref}(k) = \gamma_1(k), \tag{4.3}$$

where $\gamma_1(k)$ is an arbitrary time-varying function, and the reference for reactive power is defined as a function of electric power factor (f_{ps1}):

$$Q_s^{ref}(k) = \frac{P_s(k)}{f_{ps1}}\sqrt{1 - f_{ps1}^2}, \tag{4.4}$$

where $P_s(k) \approx \tau_e^{ref}(k)$ as explained in [38, 77].

In order to decouple the stator and rotor current and transform system

$$i_s(k+1) = i_s(k) + t_s\left(A_{11}(k)\, i_s(k) + A_{12}(k)\, i_r(k)\right) + t_s\left(D_1 v_s(k) + B_1 u(k)\right), \tag{4.5}$$

$$i_r(k+1) = i_r(k) + t_s\left(A_{21}(k)\, i_s(k) + A_{22}(k)\, i_r(k)\right) + t_s\left(D_2 v_s(k) + B_2 u(k)\right), \tag{4.6}$$

to a block control form [49], the following equalities are used:

$$i_r(k) = G_1 i_s(k) + H_1 v_s(k), \tag{4.7}$$

$$i_s(k) = G_2 i_r(k) + H_2 v_s(k), \tag{4.8}$$

with

$$G_1 = \begin{bmatrix} \frac{X_s}{X_m} & \frac{r_s}{X_m} \\ -\frac{r_s}{X_m} & \frac{X_s}{X_m} \end{bmatrix}, H_1 = \begin{bmatrix} 0 & \frac{1}{X_m} \\ -\frac{1}{X_m} & 0 \end{bmatrix},$$
$$G_2 \begin{bmatrix} \frac{X_m X_s}{r^2+X_s^2} & -\frac{X_m r_s}{r_s^2+X_s^2} \\ \frac{X_m r_s}{r_s^2+X_s^2} & \frac{X_m X_s}{r_s^2+X_s^2} \end{bmatrix}, H_2 = \begin{bmatrix} -\frac{r_s}{r^2+X_s^2} & -\frac{X_s}{r_s^2+X_s^2} \\ \frac{X_s}{r_s^2+X_s^2} & -\frac{r_s}{r_s^2+X_s^2} \end{bmatrix}.$$

Equations (4.7) and (4.8) are valid for $\forall k \in \mathbb{Z}^+ \cup 0$, where $\mathbb{Z}^+$ are all the positive integers; thus (4.5) and (4.6) can be rewritten using (4.7) and (4.8), as

$$i_s(k+1) = f_{i_s}(k) + t_s B_1 u(k), \tag{4.9}$$

$$i_r(k+1) = f_{i_r}(k) + t_s B_2 u(k), \tag{4.10}$$

where

$$\begin{aligned} fi_s(k) &= i_s(k) + t_s\left(A_{11}(k) + A_{12}(k) G_1\right) i_s(k) \\ &\quad + t_s\left(A_{12}(k) H_1 + D_1\right) v_s(k), \\ fi_r(k) &= i_r(k) + t_s\left(A_{21}(k) G_2 + A_{22}(k)\right) i_r(k) \\ &\quad + t_s\left(A_{21}(k) H_2 + D_2\right) v_s(k). \end{aligned}$$

Evaluating (4.1) at $(k+1)$ and using (4.9) and (4.10), $\tau_e(k+1)$ is defined as

$$\tau_e(k+1) = f_{\tau_e}(k) + B_{\tau_e}(k) u(k), \tag{4.11}$$

where

$$\begin{aligned} f_{\tau_e}(k) &= f_{i_r}(k)^T M_{\tau_e} f_{i_s}(k), \\ B_{\tau_e}(k) &= t_s\left(f_{i_s}(k)^T M_{\tau_e}^T B_2 + f_{i_r}(k)^T M_{\tau_e} B_1\right). \end{aligned}$$

Evaluating (4.2) at $(k+1)$ and using (4.9), then

$$Q_s(k+1) = f_{Q_s}(k) + B_{Q_s}(k) u(k), \tag{4.12}$$

where

$$\begin{aligned} f_{Q_s}(k) &= v_s(k)^T M_Q f_{i_s}(k), \\ B_{Q_s}(k) &= t_s v_s(k)^T M_Q B_1. \end{aligned}$$

From (4.11) and (4.12), the following system is formulated:

$$x_1(k+1) = f_{x_1}(k) + g_{x_1}(k) u(k), \; y(k) = x_1(k), \tag{4.13}$$

where

$$x_1(k) = \begin{bmatrix} \tau_e(k) \\ Q_s(k) \end{bmatrix}, f_{x_1}(k) = \begin{bmatrix} f_{\tau_e}(k) \\ f_{Q_s}(k) \end{bmatrix}, g_{x_1}(k) = \begin{bmatrix} B_{\tau_e}(k) \\ B_{Q_s}(k) \end{bmatrix}.$$

Clearly, (4.13) constitutes a one block system [49]; then it can be used to synthesize the control input $u(k)$ as stated in the following theorem.

Theorem 4.1

For system (4.13), the control law defined as

$$u^{c}(k) = u^{equ}(k) + u^{din}(k)$$

ensures trajectory tracking, with $u^{equ}(k)$ determined using discrete time sliding modes [80] and $u^{din}(k)$ a proportional term.

Proof. Using the discrete time sliding modes, the sliding manifold is defined as:

$$s_1(k) = x_1(k) - x_1^{ref}(k), \tag{4.14}$$

where $x_1^{ref}(k) = \begin{bmatrix} \tau_e^{ref}(k) \\ Q_s^{ref}(k) \end{bmatrix}$.

Evaluating (4.14) at $(k+1)$,

$$s_1(k+1) = f_{x_1}(k) + g_{x_1}(k)\,u(k) - x_1^{ref}(k+1), \tag{4.15}$$

the equivalent control $u^{equ}(k)$ is calculated as [80]

$$u^{equ}(k) = -g_{x_1}(k)^{-1}\left(f_{x_1}(k) - x_1^{ref}(k+1)\right). \tag{4.16}$$

The g_{x_1} determinant is defined as:

$$\begin{aligned}
|g_{x_1}| = {} & \frac{\omega_b^3 t_s^3 X_m^3}{\sigma^2 X_r^2 X_s}\left(\left(\frac{1}{\sigma}\omega_r - 1\right)\frac{1}{X_s} - \frac{X_m^2}{(r_s^2+\sigma X_s^2)X_r}\left(\omega_r + \frac{r_s^2}{X_s^2}\right)\right) i_{dr} v_{ds} \\
& + \frac{\omega_b^3 X_m^2 t_s^3}{\sigma^2 X_r^2 X_s}\left(\omega_r - 1 + \frac{r_r r_s}{\sigma X_r X_s}\right) i_{ds} v_{ds} \\
& + \frac{\omega_b^3 X_m^2 t_s^3}{\sigma^3 X_r^3 X_s^2}\left(r_r + \frac{X_m^2 r_s}{(X_s^2+r_s^2)}(\omega_r - 1)\right) v_{ds}^2 \\
& + \frac{\omega_b^2 t_s^2 X_m^3}{\sigma^2 X_r^2 X_s^2}\left(1 - \frac{\omega_b r_r t_s}{\sigma X_r} + \frac{\omega_b X_m^2 r_s t_s}{\sigma X_r (X_s^2+r_s^2)}(\omega_r - 1)\right) i_{qr} v_{ds} \\
& + \frac{\omega_b^2 X_m^2 t_s^2}{\sigma^2 X_r^2 X_s}\left(1 - \frac{\omega_b r_r t_s}{\sigma X_r} + \frac{\omega_b r_s t_s}{\sigma X_s}(\omega_r - 1)\right) i_{qs} v_{ds},
\end{aligned}$$

where $v_{ds} > 0$, $\omega_r > 0$. The numerical evaluation of (4.17), considering the parameter values given in Table 3.1, is as follows:

$$\begin{aligned}
& \left((0.44756 i_{qs} + 0.72952 i_{ds})\,\omega_r + 2.6292 i_{qs} - 0.70068 i_{ds} - 1.25 \right. \\
& \qquad \left. \times 10^{-6}\omega_r v_{ds}\right) v_{ds} \\
& -0.6029 v_{ds}.
\end{aligned}$$

This determinant is zero only when the DFIG is not energized ($i_{ds} = i_{qs} = i_{dr} = i_{qr} = v_{ds} = 0$) or $i_{ds} = -0.19711$, $i_{ds} = 0.12092$, where this is only possible when the DFIG is working as a motor. Then, the inverse of g_{x_1} for this particular case always exists when the DFIG is working as a generator.

Applying $u(k) = u^{equ}(k)$ to (4.15), the state of the closed-loop system reaches the sliding manifold $s_1(k) = 0$ in one sample time. However, it is appropriate to add to the control signal a stabilizing term $u^{din}(k)$, defined as

$$u^{din}(k) = g_{x_1}(k)^{-1}\left(K_s s_1(k) + K_0 s_0(k)\right), \tag{4.17}$$

where $K_s s(k)$ is included in order to reach the sliding surface asymptotically and to avoid high gain control, and $K_0 s_0(k)$ is an integral term to reject the unmodeled dynamics and to reduce the steady-state error. The integral term $s_0(k)$ is defined as

$$s_0(k+1) = s_0(k) + t_s s_1(k). \tag{4.18}$$

Hence, $u^c(k)$ is proposed as

$$u^c(k) = u^{equ}(k) + u^{din}(k). \tag{4.19}$$

Applying $u(k) = u^c(k)$ in (4.15), the close-loop sliding manifold dynamics is

$$s_1(k+1) = K_s s_1(k) + K_0 s_0(k). \tag{4.20}$$

Using (4.18) and (4.20), the following subsystem is obtained:

$$\begin{bmatrix} s_0(k+1) \\ s_1(k+1) \end{bmatrix} = \begin{bmatrix} I_2 & t_2 I_2 \\ K_0 & K_s \end{bmatrix} \begin{bmatrix} s_0(k) \\ s_1(k) \end{bmatrix}, \tag{4.21}$$

then K_s, K_0 should be selected such that $\begin{bmatrix} I_2 & t_s I_2 \\ K_0 & K_s \end{bmatrix}$ is a Schur matrix [81], where I_2 is a $2x2$ identity matrix.

To take into account the boundedness of the control signal $\|u(k)\| < u_{\max}$, $u_{\max} > 0$, where $\|\cdot\|$ stands for the Euclidean norm, the following control law is selected [80]:

$$u(k) = \begin{cases} u_{\max}\frac{u^c(k)}{\|u^c(k)\|} & if \|u^c(k)\| > u_{\max} \\ u^c(k) & if \|u^c(k)\| \le u_{\max} \end{cases}. \tag{4.22}$$

The stability proof using (4.22) is presented in [80].

4.1.1.1 Simulation Results

To evaluate the performance of the proposed controller, a simulation for a three-phase generator with a stator-referred rotor is implemented. The generator parameters appear in Table 3.1. The simulation is performed in MATLAB/Simulink®[1]. The simulation conditions are:

Simulation time: 10 seconds.

Sampling time: $t_s = 0.5\ ms$.

DFIG initial conditions: rotor speed 0.3 *pu*, $i_{ds} = 0.001\ pu$, $i_{qs} = 0.001\ pu$, $i_{dr} = 0.001\ pu$, $i_{qr} = 0.001\ pu$.

The initial electric torque reference is a constant signal in 0.4 *pu*.

From 1 to 3 seconds a pulse variation in the electric torque reference with amplitude of 0.5 *pu* is incepted.

At 5 seconds the electric torque reference is changed to a senoidal signal centered at 0.5 *pu* with amplitude of 0.4 *pu* and 1 *Hz*.

Power factor reference is constant in 0.9.

The gain K_s in (4.20) is defined as $\begin{bmatrix} 0.8 & 0 \\ 0 & 0.8 \end{bmatrix}$.

The gain K_0 in (4.20) is defined as $\begin{bmatrix} -20 & 0 \\ 0 & -20 \end{bmatrix}$.

The performance of the controller is illustrated in Figures 4.1 to 4.4. The electric torque tracking is illustrated in Figure 4.1(a), where it can be seen that the electric torque (τ_e) reaches the reference $\left(\tau_e^{ref}\right)$ quickly. The reactive power tracking is shown in Figure 4.1(b), where the $\left(Q_s^{ref}\right)$ is calculated to keep the electric power factor constant at 0.9. In Figure 4.1(c), we can see that an abrupt change in the electric torque reference does not affect the electric power factor (f_{ps1}). The stator and rotor currents are bounded as shown in Figure 4.2, where the current performance is similar to the electric torque reference. The control signals are bounded as shown in Figure 4.3. Due to the fact that the electric torque tracking error is small, the rotor speed is almost constant as shown in Figure 4.4.

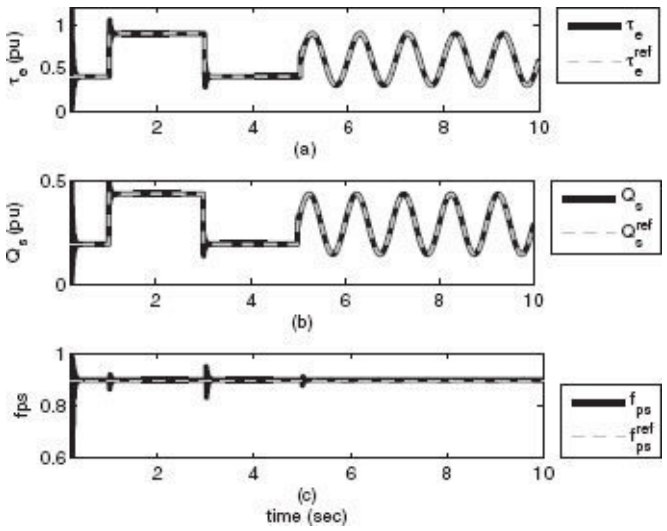

Figure 4.1 System outputs: (a) electric torque (τ_e) tracking, (b) reactive power (Q_s) tracking, and (c) power factor (f_{ps1}) tracking.

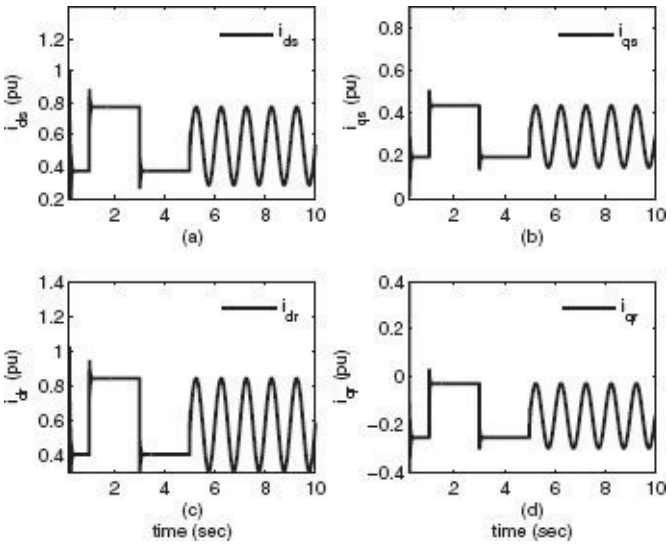

Figure 4.2 Generator currents: (a) stator current (i_{ds}), (b) stator current (i_{qs}), (c) rotor current (i_{dr}), and (d) rotor current (i_{qr}).

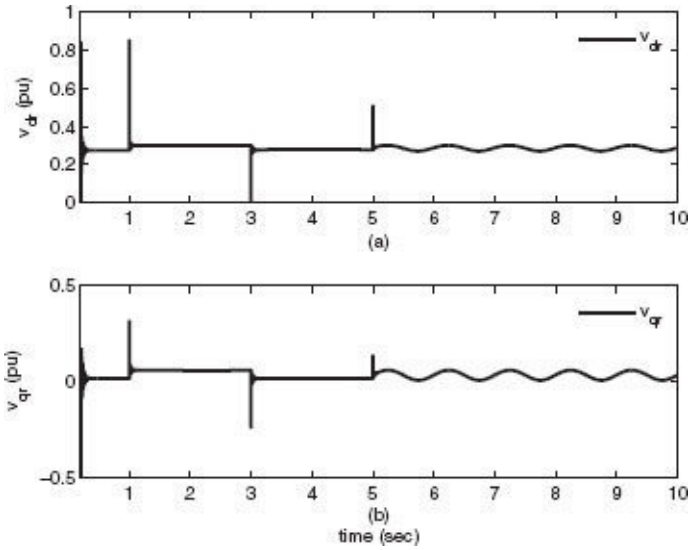

Figure 4.3 Control signals: (a) v_{dr} and (b) v_{qr}.

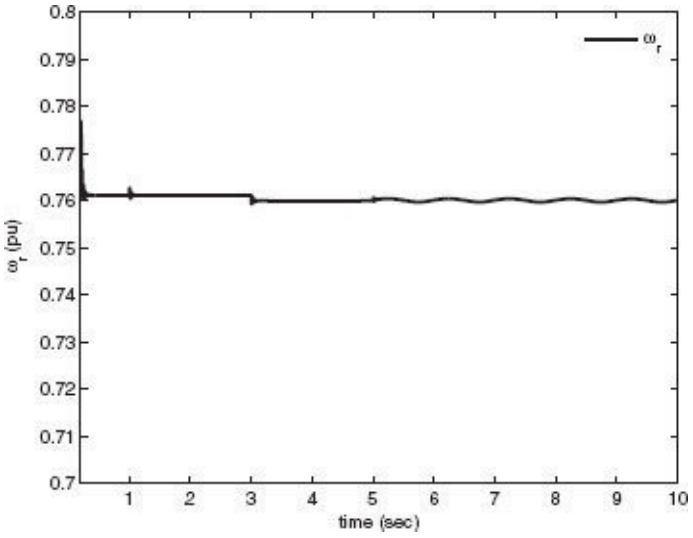

Figure 4.4 Rotor speed (ω_r).

4.1.2 DC LINK CONTROLLER

In this section the GSC controller using sliding modes is developed. As mentioned in Section 3.2.2, the GSC works like a rectifier to keep the DC voltage constant in the DC Link. Hence, the variables to be controlled are the capacitor voltage ($v_{dc}(k)$) and the reactive power ($Q_g(k)$). The control objectives are: a) to track a DC voltage reference $\left(v_{dv}^{ref}(k)\right)$ on the DC Link, and b) to keep constant the electric power factor ($f_{ps2}(k)$) at the step-up transformer terminals through the reactive power ($Q_g(k)$) control.

The step-up transformer reactive power ($Q_g(k)$) is defined as

$$Q_g(k) = v_{gs}(k)^T M_Q i_g(k). \tag{4.23}$$

The DC voltage reference $\left(v_{dc}^{ref}(k)\right)$ is defined as

$$v_{dc}^{ref}(k) = \gamma_2(k), \tag{4.24}$$

where $\gamma_2(k)$ is an arbitrary time-varying function, and the reference for the reactive power is defined as a function of the electric power factor (f_{ps2}) [38, 77]:

$$Q_g^{ref}(k) = \frac{P_g(k)}{f_{ps2}}\sqrt{1 - f_{ps2}^2}, \tag{4.25}$$

where

$$P_g(k) = v_{gs}^T M_P i_g(k), \tag{4.26}$$

with $M_P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Let us define the tracking error for the DC voltage as

$$\varepsilon_1^g(k) = v_{dc}(k) - v_{dc}^{ref}(k). \tag{4.27}$$

From (4.27), using (4.28)

$$v_{dc}(k+1) = v_{dc}(k) + t_s\left(\frac{1}{Cv_{dc}(k)} v_{gs}^T(k) M_P i_g(k)\right), \tag{4.28}$$

then $\varepsilon_1^g(k+1)$ is equal to

$$\varepsilon_1^g(k+1) = v_{dc}(k) + t_s\left(\frac{1}{Cv_{dc}(k)} v_{gs}^T(k) M_P i_g(k)\right) - v_{dc}^{ref}(k+1), \tag{4.29}$$

and considering that in the dq axes, $v_{qg}(k) = 0$, the DC voltage is controlled directly by $i_{dg}(k)$.

$$\varepsilon_1^g(k+1) = v_{dc}(k) + \frac{t_s}{Cv_{dc}(k)} v_{dgs}(k) i_{dg}(k) - v_{dc}^{ref}(k+1). \tag{4.30}$$

Then, the i_{dg} reference is defined as

$$i_{dg}^{ref}(k) = \frac{Cv_{dc}(k)}{t_s v_{ds}(k)}\left(v_{dc}^{ref}(k+1) - v_{dc}(k) + k_1\varepsilon_1^g(k) + k_0\varepsilon_0^g(k)\right), \tag{4.31}$$

where, by a similar procedure to obtain (4.21), using (4.31) and (4.32), k_0 and k_1 are selected such that $\begin{bmatrix} 1 & t_s \\ k_0 & k_1 \end{bmatrix}$ is a Schur matrix [81], and $k_1\varepsilon_1^g(k)$ is introduced to reach the reference asymptotically; in order to reject unmodeled dynamics and to reduce the steady-state error, an integral term $k_0\varepsilon_0^g(k)$ is inserted as

$$\varepsilon_0^g(k+1) = \varepsilon_0^g(k) + t_s\varepsilon_1^g(k). \tag{4.32}$$

On the other hand, the tracking error for the reactive power is

$$\varepsilon_2^g(k) = Q_g(k) - Q_g^{ref}(k). \tag{4.33}$$

From (4.23), and considering that $v_{qgs}(k) = 0$, it can be established that

$$Q_g(k) = -v_{dgs}(k) i_{qg}(k). \tag{4.34}$$

In order to determine the reference i_{qg}^{ref}, we assume that $\varepsilon_2^g(k) = 0$, which implies $Q_g(k) = Q_g^{ref}(k)$; hence:

$$i_{qg}^{ref}(k) = -i_{dg}(k)\frac{\sqrt{1-f_{ps2}^2}}{f_{ps2}}. \tag{4.35}$$

Considering, $x_2(k) = i_g(k)$, equation

$$i_g(k+1) = i_g(k) + t_s\left(A_g i_g(k) + B_g v_{gs}(k) - B_g u_g(k)\right) \tag{4.36}$$

can be rewritten as follows:

$$x_2(k+1) = f_{x_2}(k) + g_{x_2} u_g(k) \tag{4.37}$$

with

$$f_{x_2}(k) = i_g(k) + t_s\left(A_g i_g(k) + B_g v_{gs}(k)\right),\ g_{x_2} = -t_s B_g.$$

Then, it is clear that equation (4.37) is in the form of (4.13); according to Theorem 4.1, the control input $u_g(k)$ is selected as follows.

The sliding manifold is formulated as

$$s_g(k) = x_2(k) - x_2^{ref}(k), \tag{4.38}$$

where $x_2^{ref}(k) = \begin{bmatrix} i_{dg}^{ref}(k) \\ i_{qg}^{ref}(k) \end{bmatrix}$.

Evaluating (4.38) at $(k + 1)$, the equivalent control $\left(u_g^{equ}(k)\right)$ is calculated as [80]

$$u_g^{equ}(k) = -\frac{1}{t_s} g_{x_2}^{-1}\left(f_{x_2}(k) - x_2^{ref}(k+1)\right). \tag{4.39}$$

The matrix g_{x_2} is a constant diagonal matrix $g_{x_2} = \begin{bmatrix} \frac{\omega_b}{X_l} & 0 \\ 0 & \frac{\omega_b}{X_l} \end{bmatrix}$; its determinant is not zero for $\forall k \in \mathbb{Z}+ \cup 0$. Then the g_{x_2} inverse always exists.

Applying $u(k) = u_g^{equ}(k)$ to (4.37), the state of the closed-loop system reaches the sliding manifold $s_g(k) = 0$ in one sample time. However, it is appropriate to add to the control signal a stabilizing term $u_g^{din}(k)$ in order to reach the sliding surface asymptotically and to avoid high gain control; hence, the $u_g^c(k)$ is determined as

$$u_g^c(k) = u_g^{equ}(k) + u_g^{din}(k), \tag{4.40}$$

where

$$u_g^{din}(k) = \frac{1}{t_s} g_{x_2}^{-1} \left(K_g s_g(k) + K_{0_g} s_{0_g}(k) \right), \tag{4.41}$$

$K_g = \begin{bmatrix} k_1^g & 0 \\ 0 & k_2^g \end{bmatrix}, K_{0_g} = \begin{bmatrix} 0 \\ k_0^g \end{bmatrix}$; in order to reject the unmodeled dynamics and to reduce the steady-state error, the integral term $s_{0_g}(k)$ is inserted, as

$$s_{0_g}(k+1) = s_{0_g}(k) + t_s \left(i_{qg}(k) - i_{qg}^{ref}(k) \right), \tag{4.42}$$

and $\begin{bmatrix} k_1^g & 0 & 0 \\ 0 & 1 & t_s \\ 0 & k_0^g & k_2^g \end{bmatrix}$ is a Schur matrix [81].

To take into account the boundedness of the control signal $\left\| u_g(k) \right\| < u_{g\,\max}$, $u_{g\,\max} > 0$, where $\left\| \cdot \right\|$ stands for the Euclidean norm, the following control law is selected [80]:

$$u_g(k) = \begin{cases} u_g \max \frac{u_g^c(k)}{\|u_g^c(k)\|} & if \|u_g^c(k)\| > u_{g\,\max} \\ u_g^c(k) & if \|u_g^c(k)\| \leq u_{g\,\max} \end{cases}. \tag{4.43}$$

The stability proof using (4.43) is presented in [80].

4.1.2.1 Simulation Results

To evaluate the performance of the proposed controller a simulation for a DC Link is developed. The DC Link parameters appear in Table 3.1. The simulation is performed in MATLAB/Simulink®[2], with conditions:

Simulation time: 5 seconds.

Sampling time: $t_s = 0.5\ ms$.

DC Link initial conditions: $v_{dc} = 0.01\ pu$, $i_{dg} = 0\ pu$, $i_{qg} = 0\ pu$.

The DC voltage reference is a constant signal at 0.5567 *pu*.

Power factor reference is constant at 0.9.

Control inputs are incepted at 0 seconds.

Load resistance R_L of 383.0579 *pu* connected to the capacitor in parallel scheme in order to simulate an unknown perturbation.

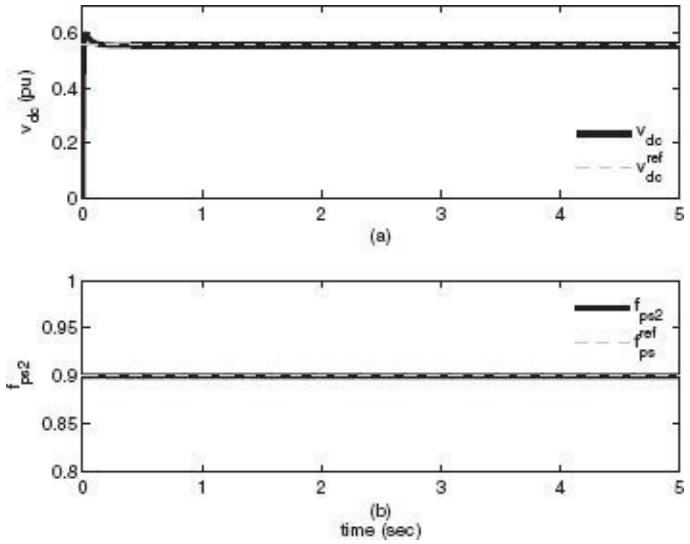

Figure 4.5 System outputs: (a) DC voltage (v_{dc}) and (b) step-up transformer power factor (f_{ps2}).

The performance of the GSC controller is shown in Figures 4.5 to 4.7. The DC voltage (Figure 4.5(a)) and power factor tracking (Figure 4.5(b)) are controlled to the reference. In this figure, we can see that the DC voltage reached the reference quickly, and the electric power factor kept constant at 0.9 all the time. The control signals v_{dg} and v_{qg} are bounded as shown in Figure 4.6. The state variables of the DC Link (3.9) and (3.10) are shown in Figure 4.7, where we can see that the state variables are stable and bounded. The transient time is short and the resistance R_L is connected in parallel scheme to simulate the capacitor discharge by a resistive load. The controller presents robustness against resistive loads (R_L) in the range of $1M\Omega$ to 100Ω; if the resistance value is below this range the controller is saturated and the DC voltage tracking is not possible.

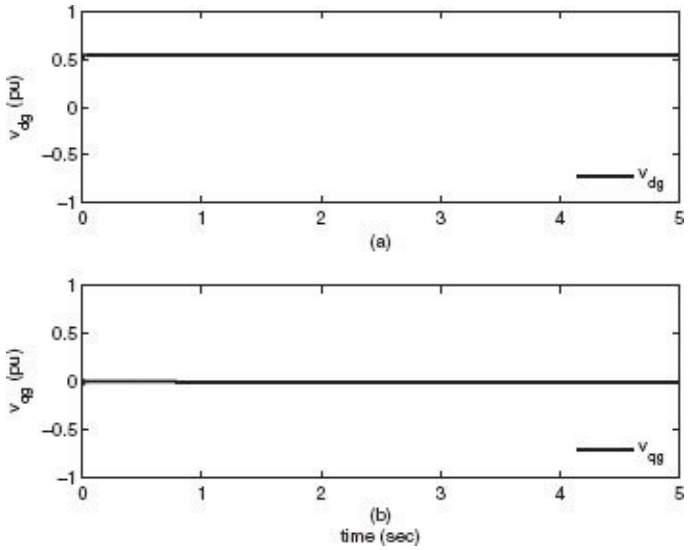

Figure 4.6 Control signals: (a) v_{dg} and (b) v_{qg}.

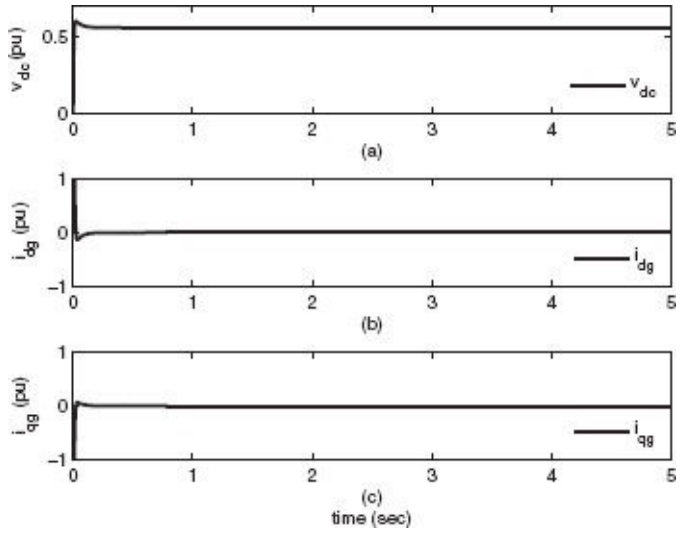

Figure 4.7 State variables: (a) DC voltage v_{dc}, (b) current i_{dg}, and (c) current i_{qg}.

4.2 INVERSE OPTIMAL CONTROL

Similarly to Section 4.1, as a first step a controller is designed based on inverse optimal control algorithm for DFIG; after that, the simulation results are shown. Then, a GSC controller is designed under the same scheme and the corresponding simulation results are presented.

4.2.1 DFIG CONTROLLER

The variables to be controlled are the DFIG electric torque ($\tau_e(k)$) and the stator reactive power ($Q_s(k)$). The control objectives are: a) to track an electric torque trajectory $\left(\tau_e^{ref}(k)\right)$, and b) to keep the electric power factor ($f_{ps1}(k)$) at the stator terminals constant by means of the stator reactive power control. The electric torque ($\tau_e(k)$) and stator reactive power ($Q_s(k)$) are defined, respectively, as (4.1) and (4.2).

In order to apply the inverse optimal control, we need to calculate the references for the system state, which are obtained by considering the steady-state as the desired ones, and then achieving trajectory tracking as a stabilization problem; the steady-state is obtained as follows.

The electric torque reference $\left(\tau_e^{ref}(k)\right)$ is defined as

$$\tau_e^{ref}(k) = \gamma_1(k), \tag{4.44}$$

where $\gamma_1(k)$ is an arbitrary time-varying function, and the reactive power reference is defined as a function of electric power factor (f_{ps1}) [38, 77]:

$$Q_s^{ref}(k) = \frac{P_s(k)}{f_{ps1}}\sqrt{1 - f_{ps1}^2}, \tag{4.45}$$

where $P_s(k) \approx \tau_e^{ref}(k)$.

The tracking errors are defined, respectively, as

$$\varepsilon_{\tau_e}(k) = \tau_e(k) - \tau_e^{ref}(k), \tag{4.46}$$

$$\varepsilon_{Q_s}(k) = Q_s(k) - Q_s^{ref}(k). \tag{4.47}$$

In order to calculate the steady-state, it is assumed that $\varepsilon_{\tau_e}(k) = 0$ and $\varepsilon_{Q_s}(k) = 0$; then using (4.1) and (4.2), equations (4.46) and (4.47) can be rewritten as

$$i_r^{ss}(k)^T M_{\tau_e} i_s^{ss}(k) = \tau_e^{ref}(k), \tag{4.48}$$

$$v_s(k)^T M_{Q_s} i_s^{ss}(k) = Q_s^{ref}(k), \tag{4.49}$$

where the superscript *ss* denotes steady-state. From (4.5) and (4.6), the relation between stator and rotor currents in steady-state is obtained as [69]:

$$i_r^{ss}(k) = G_1 i_s^{ss}(k) + H_1 v_s(k), \tag{4.50}$$

$$i_s^{ss}(k) = G_2 i_r^{ss}(k) + H_2 v_s(k), \tag{4.51}$$

where

$$G_1 = \begin{bmatrix} \frac{X_s}{X_m} & \frac{r_s}{X_m} \\ -\frac{r_s}{X_m} & \frac{X_s}{X_m} \end{bmatrix}, H_1 = \begin{bmatrix} 0 & \frac{1}{X_m} \\ -\frac{1}{X_m} & 0 \end{bmatrix},$$
$$G_2 = \begin{bmatrix} \frac{X_m X_s}{r_s^2+X_s^2} & -\frac{X_m r_s}{r_s^2+X_s^2} \\ \frac{X_m r_s}{r_s^2+X_s^2} & \frac{X_m X_s}{r_s^2+X_s^2} \end{bmatrix}, H_2 = \begin{bmatrix} -\frac{r_s}{r_s^2+X_s^2} & -\frac{X_s}{r_s^2+X_s^2} \\ \frac{X_s}{r_s^2+X_s^2} & -\frac{r_s}{r_s^2+X_s^2} \end{bmatrix}.$$

Substituting (4.50) in (4.48) and reducing terms, the following equation is obtained:

$$r_s i_{ds}^{ss}(k)^2 + r_s i_{qs}^{ss}(k)^2 + i_{ds}^{ss}(k) v_{ds}(k) + i_{qs}^{ss}(k) v_{qs}(k) = \tau_e^{ref}(k). \tag{4.52}$$

It is required to know $i_{ds}^{ss}(k)$ and $i_{qs}^{ss}(k)$, then solving (4.49) for $i_{qs}^{ss}(k)$,

$$i_{qs}^{ss}(k) = \frac{Q_s^{ref}(k)}{v_{ds}(k)} + i_{ds}^{ss}(k) \frac{v_{qs}(k)}{v_{ds}(k)}. \tag{4.53}$$

Substituting (4.53) in (4.52) the following equation is obtained:

$$a_{ss} i_{ds}^{ss}(k)^2 + b_{ss} i_{ds}^{ss}(k) + c_{ss} = 0, \tag{4.54}$$

where

$$a_{ss} = r_s + r_s \frac{v_{qs}(k)^2}{v_{ds}(k)^2},$$
$$b_{ss} = v_{ds} + \frac{v_{qs}(k)^2}{v_{ds}(k)} + Q_s^{ref}(k)\, r_s \frac{v_{qs}(k)}{v_{ds}(k)^2},$$
$$c_{ss} = \frac{Q_s^{ref}(k)}{v_{ds}(k)} v_{qs}(k) + \frac{r_s Q_s^{ref}(k)^2}{v_{ds}(k)^2} - \tau_e^{ref}(k).$$

A solution of (4.54) for $i_{ds}^{ss}(k)$ is given by

$$i_{ds}^{ss}(k) = \frac{-b_{ss} + \sqrt[2]{b_{ss}^2 - 4a_{ss}c_{ss}}}{2a_{ss}} \tag{4.55}$$

Using (4.53), (4.55), and $i_r^{ss}(k)$ given by (4.50), then $i^{ss}(k)$ is defined as

$$i^{ss}(k) = \begin{bmatrix} i_{ds}^{ss}(k) \\ i_{qs}^{ss}(k) \\ i_{dr}^{ss}(k) \\ i_{qr}^{ss}(k) \end{bmatrix}. \tag{4.56}$$

The system denoted by (3.6) and (3.7) is rewritten as

$$i(k+1) = (I_4 + A_t(k))\, i(k) + D_t v_s(k) + B_t u(k), \tag{4.57}$$

where I_4 is the 4×4 identity matrix,

$$A_t(k) = t_s \begin{bmatrix} A_{11}(k) & A_{12}(k) \\ A_{21}(k) & A_{22}(k) \end{bmatrix}, D_t = t_s \begin{bmatrix} D_1 \\ D_2 \end{bmatrix}, B_t = t_s \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}.$$

In order to apply the inverse optimal control, the error tracking is defined as

$$i_e(k) = i(k) - i^{ss}(k). \tag{4.58}$$

Evaluating (4.58) at time $(k + 1)$ and using (3.6), the error dynamics is given by

$$\begin{aligned} i_e(k+1) &= i(k+1) - i^{ss}(k+1), \\ i_e(k+1) &= (I_4 + A_t(k))\, i(k) + D_t v_s(k) + B_t u(k) \\ &\quad - i^{ss}(k+1) \\ i_e(k+1) &= (I_4 + A_t(k))\, i_e(k) + (I_4 + A_t(k))\, i^{ss}(k) \\ &\quad + B_t u(k) + D_t v_s(k) - i^{ss}(k+1). \end{aligned} \tag{4.59}$$

For system (4.59), the control signal $u(k)$ is decomposed into two components:

$$u(k) = u_1(k) + u_i^*(k). \tag{4.60}$$

$u_1(k)$ is selected as

$$\begin{aligned} u_1(k) = \; & B_t^{*-1}(-(I_4 + A_t(k))\, i^{ss}(k) - D_t v_s(k) \\ & + i^{ss}(k+1)), \end{aligned} \tag{4.61}$$

where $B_t^{*-1} = \left(B_t^T B_t\right)^{-1} B_t^T$. Then, system (4.59), with (4.60) as input, results in

$$i_e(k+1) = (I_4 + A_t(k))\, i_e(k) + B_t u_i^*(k) \tag{4.62}$$

which has the form of (2.35) as

$$x(k+1) = f_i(x(k)) + g_i(x(k))\, u_i^*(k),$$

where $x(k) := i_e(k)$ is the system state, $f_i(x(k)) = (I_4 + A_t(k))\, i_e(k)$, $g_i(x(k)) = B_t$, and hence the inverse optimal control law $u_i^*(k)$ is established by Theorem 2.1 as

$$\begin{aligned} u_i^*(k) = \; & -\tfrac{1}{2}\left(R_1(x(k)) + \tfrac{1}{2} g_i^T(x(k))\, P_1\right. \\ & \left. \times g_i(x(k))\right)^{-1} g_i^T(x(k))\, P_1 f_i(x(k)), \end{aligned} \tag{4.63}$$

where for the RSC controller, $R_1(x(k))$ and P_1 are the $R(x(k))$ and P matrices in (2.45), respectively, which are symmetric and positive definite matrices.

The choice of the matrix P_1 in (4.63) is very important, because it determines the performance of the controller. For this reason, in Appendix A, a heuristic optimization algorithm is proposed to determine the optimal value for the matrix P_1 for the output reference tracking.

4.2.1.1 Simulation Results

To evaluate the performance of the developed controller a simulation is performed. The DFIG is a three-phase generator, four poles, with a stator-referred rotor. The generator parameters appear in Table 3.1. The simulation conditions are:

Simulation time: 10 seconds.

Sampling time: $t_s = 0.5\ ms$.

DFIG initial conditions: rotor speed 0.3 *pu*, $i_{ds} = 0.001\ pu$, $i_{qs} = 0.001\ pu$, $i_{dr} = 0.001\ pu$, $i_{qr} = 0.001\ pu$.

The initial electric torque reference is a constant signal of 0.4 *pu*.

From 1 to 3 seconds a pulse variation in the electric torque reference with amplitude of 0.5 *pu* is incepted.

At 5 seconds the electric torque reference is changed to a senoidal signal centered at 0.5 *pu* with amplitude of 0.4 *pu* and 1 *Hz*.

Power factor reference is constant at 0.9.

The matrix P_1 in (4.63) is defined as

$$\begin{bmatrix} 4.805 & 35.679 & 0.392 & -38.965 \\ 35.679 & -39.157 & -0.634 & 10.695 \\ 0.392 & -0.634 & 5.027 & -0.566 \\ -38.965 & 10.695 & -0.566 & 21.098 \end{bmatrix}.$$

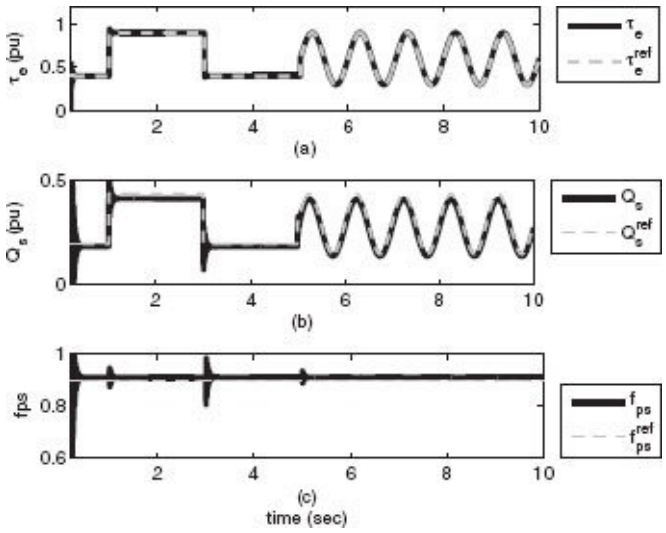

Figure 4.8 System outputs: (a) electric torque (τ_e) tracking, (b) reactive power (Q_s) tracking, and (c) power factor (f_{ps1}) tracking.

Figure 4.8 shows the electric torque (τ_e) tracking, the reactive power (Q_s) tracking, and the power factor (f_{ps1}) controlled to the power factor constant of 0.9. It is easy to see that the transient lapse is short and the reference signals are reached in less than 0.1 seconds. The reactive power tracking has a steady-state error, Figure 4.8(b); despite this error, the power factor (f_{ps1}) is kept very close to the desired value 0.9 as shown in Figure 4.8(c). As expected, i_{ds}, i_{qs}, i_{dr}, and i_{qr} converge closely to their steady-state reference; these currents are presented in Figure 4.9. The tracking error for the reactive power (Q_s) is caused by the steady-state errors of the rotor currents i_{dr} and i_{qr} as displayed in Figure 4.9(b) and Figure 4.9(d). The control signals obtained by the inverse optimal control technique are bounded, as shown in Figure 4.10. The rotor speed (ω_r) decreases slowly due to the steady-state error in the rotor currents. The rotor speed is shown in Figure 4.11.

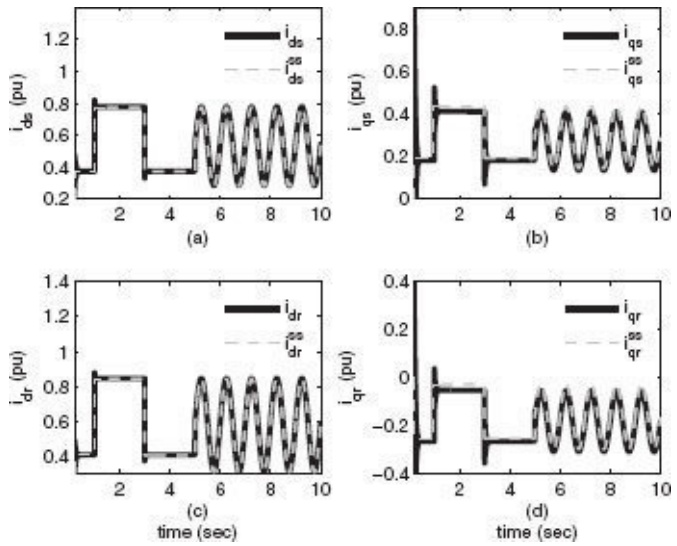

Figure 4.9 Generator currents: (a) stator current i_{ds}, (b) stator current i_{qs}, (c) rotor current i_{dr}, and (d) rotor current i_{qr}.

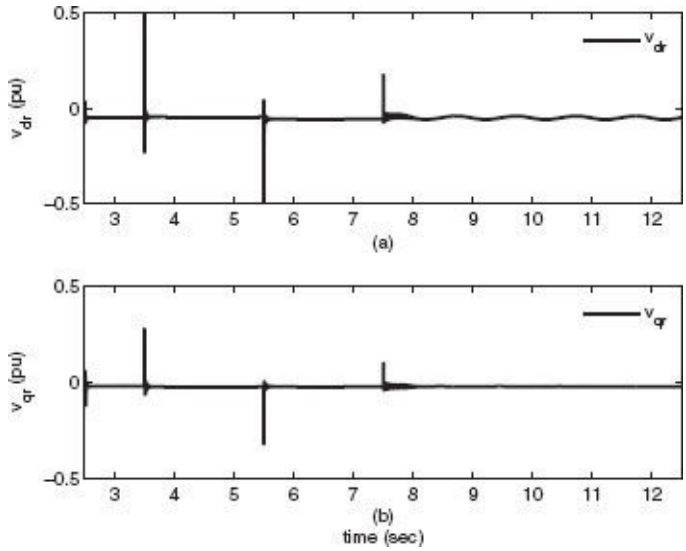

Figure 4.10 Control signals: (a) v_{dr} and (b) v_{qr}.

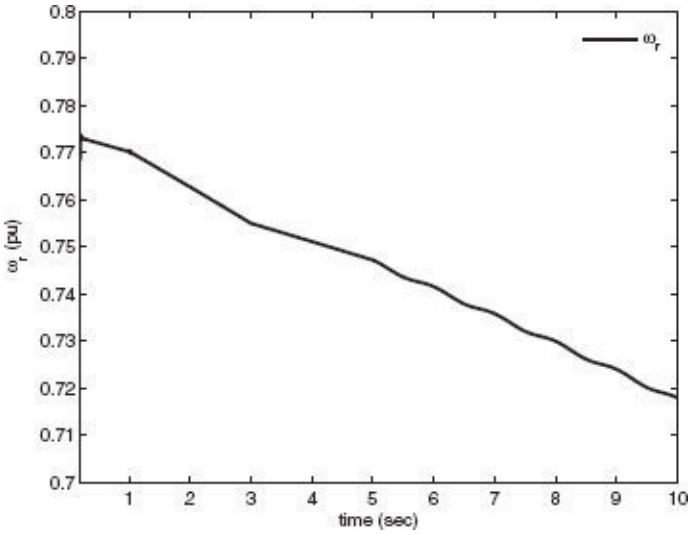

Figure 4.11 Rotor speed ω_r.

4.2.2 DC LINK CONTROLLER

The variables to be controlled are the capacitor voltage ($v_{dc}(k)$) and the reactive power ($Q_g(k)$). The control objectives are: a) to track a DC voltage reference $\left(v_{dc}^{ref}(k)\right)$ for the DC Link, and b) to keep the electric power factor ($f_{ps2}(k)$) constant at the step-up transformer terminals by means of the reactive power control ($Q_g(k)$).

The step-up transformer reactive power ($Q_g(k)$) is defined as

$$Q_g(k) = v_{gs}(k)^T M_Q i_g(k). \tag{4.64}$$

The DC voltage reference $\left(v_{dc}^{ref}(k)\right)$ is defined as

$$v_{dc}^{ref}(k) = \gamma_2(k), \tag{4.65}$$

where $\gamma_2(k)$ is an arbitrary time-varying function, and the reference for the reactive power is defined as a function of electric power factor (f_{ps2}) as [38, 77]

$$Q_g^{ref}(k) = \frac{P_g(k)}{f_{ps2}}\sqrt{1 - f_{ps2}^2}, \tag{4.66}$$

where

$$P_g(k) = v_{gs}^T M_p i_g(k), \tag{4.67}$$

with $M_P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Similar to the previous case (RSC controller), it is required to determine the steady-state values for the v_{dc}^{ref} and Q_g^{ref}. The tracking error for the DC voltage and the reactive power are defined, respectively, as

$$\varepsilon_{v_{dc}}(k) = v_{dc}(k) - v_{dc}^{ref}(k), \tag{4.68}$$

$$\varepsilon_{Q_g}(k) = Q_g(k) - Q_g^{ref}(k). \tag{4.69}$$

Evaluating (4.68) at time ($k + 1$), and using (4.28), yields

$$\varepsilon_{v_{dc}}(k+1) = v_{dc}(k) + \frac{t_s v_{gs}^T(k) M_P}{C v_{dc}(k)} i_g(k) - v_{dc}^{ref}(k+1). \tag{4.70}$$

Assuming that in steady-state $\varepsilon_{v_{dc}}(k) = \varepsilon_{v_{dc}}(k+1) = \varepsilon_{Q_g}(k) = \varepsilon_{Q_g}(k+1) = 0$, then $v_{dc}^{ss}(k) = v_{dc}^{ref}(k)$, and then (4.70) is rewritten as

$$v_{dc}^{ss}(k)+\frac{t_s v_{gs}^T(k) M_P}{C v_{dc}^{ss}(k)} i_g^{ss}(k)-v_{dc}^{ref}(k+1)=0. \tag{4.71}$$

From (4.69) using (4.64), we define

$$\begin{aligned} Q_g(k)-Q_g^{ref}(k)&=0,\\ v_{gs}(k)^T M_Q i_g^{ss}(k)-P_g(k)\frac{\sqrt{1-f_{ps2}^2}}{f_{ps2}}&=0,\\ v_{gs}(k)^T M_Q i_g^{ss}(k)-v_{gs}^T M_P i_g^{ss}(k)\frac{\sqrt{1-f_{ps2}^2}}{f_{ps2}}&=0. \end{aligned} \tag{4.72}$$

From (4.71) and (4.72), the following equation is obtained:

$$A^*(k) i_g^{ss}(k)=B^*(k), \tag{4.73}$$

where

$$A^*(k)=\begin{bmatrix} \frac{t_s}{C v_{dc}^{ss}(k)} v_{gs}^T(k) M_P \\ v_{gs}(k)^T M_Q-v_{gs}^T M_P \frac{\sqrt{1-f_{ps2}^2}}{f_{ps2}} \end{bmatrix},$$

$$B^*(k)=\begin{bmatrix} v_{dc}^{ref}(k+1)-v_{dc}^{ss}(k) \\ 0 \end{bmatrix}.$$

Then $i_g^{ss}(k)$ is calculated as

$$i_g^{ss}(k)=A^*(k)^{-1}B^*(k). \tag{4.74}$$

Defining the new variables $x_g(k) = [v_{dc}(k) \quad i_{dg}(k) \quad i_{qg}(k)]^T$, $x_g^{ss}(k)=\left[v_{dc}^{ss}(k) i_{dg}^{ss}(k) i_{qg}^{ss}(k)\right]^T$, the tracking error is written as

$$\varepsilon_{x_g}(k)=x_g(k)-x_g^{ss}(k). \tag{4.75}$$

Evaluating (4.75) at time (k + 1), the tracking error dynamics is obtained as

$$\varepsilon_{x_g}(k+1)=f_\varepsilon(k)+g_\varepsilon u_g(k)+h_\varepsilon(k), \tag{4.76}$$

with

$$f_{\varepsilon}(k)=\begin{bmatrix} v_{dc}(k)+\beta v_{gs}^{T}(k) M_{P} i_{g}(k) \\ i_{g}(k)+t_{s} A_{g} i_{g}(k) \end{bmatrix},$$
$$h_{\varepsilon}(k)=\begin{bmatrix} 0 \\ t_{s} B_{g} v_{gs}(k) \end{bmatrix},$$
$$g_{\varepsilon}=\begin{bmatrix} 0 \\ -t_{s} B_{g} \end{bmatrix}, \beta=\frac{t_{s}}{C^{*} v_{dc}(k)}.$$

Then the control signal $u_g(k)$ is decomposed in two components as

$$u_{g}(k)=u_{2}(k)+u_{g}^{*}(k), \tag{4.77}$$

where

$$u_{2}(k)=g_{\varepsilon}^{*-1}\left(-f_{\varepsilon}(k)-h_{\varepsilon}(k)+f_{\varepsilon}^{*}(k)\right) \tag{4.78}$$

with $g_{\varepsilon}^{*-1}=\left(g_{\varepsilon}^{T} g_{\varepsilon}\right)^{-1} g_{\varepsilon}^{T}$.

System (4.76), with (4.77) as input, results in

$$\varepsilon_{x_{g}}(k+1)=f_{\varepsilon}^{*}(k)+g_{\varepsilon} u_{g}^{*}(k), \tag{4.79}$$

with

$$f_{\varepsilon}^{*}(k)=\begin{bmatrix} \varepsilon_{v_{dc}}(k)+\beta v_{gs}^{T}(k) M_{P} \varepsilon_{i_{g}}(k) \\ \varepsilon_{i_{g}}(k)+t_{s} A_{g} \varepsilon_{i_{g}}(k) \end{bmatrix}. \tag{4.80}$$

System (4.79) is of the form (2.35); hence the proposed inverse optimal control law $u_g^*(k)$ by Theorem 2.1 becomes

$$\begin{aligned} u_{g}^{*}(k)= & -\tfrac{1}{2}\left(R_{2}(x(k))+\tfrac{1}{2} g_{\varepsilon}^{T} P_{2} g_{\varepsilon}\right)^{-1} \\ & \times g_{\varepsilon}^{T} P_{2} f_{\varepsilon}^{*}(k), \end{aligned} \tag{4.81}$$

where for the GSC controller, $R_2(x(k))$ and P_2 are the $R(x(k))$ and P matrices in (2.45), respectively, which are symmetric and positive definite matrices.

The choice of the matrix P_2 in (4.81) is very important, because it determines the performance of the controller. For this reason, in Appendix A, a heuristic optimization algorithm is proposed to determine the optimal value for the matrix P_2 for the output reference tracking.

4.2.2.1 Simulation Results

To evaluate the performance of the proposed controller a simulation for a DC Link is developed, with parameters as in Table 3.1 and conditions:

Simulation time: 10 seconds.

Sampling time: $t_s = 0.5\ ms$.

DC Link initial conditions: $v_{dc}(0) = 0.01\ pu$, $i_{dg}(0) = i_{qg}(0) = 0\ pu$.

The DC voltage reference is a constant signal at 0.5567 *pu*.

Control inputs are incepted at 0 seconds.

Constant desired power factor: 0.9.

Load resistance R_L of 383.0579 *pu* connected to the capacitor in parallel scheme in order to simulate an unknown perturbation.

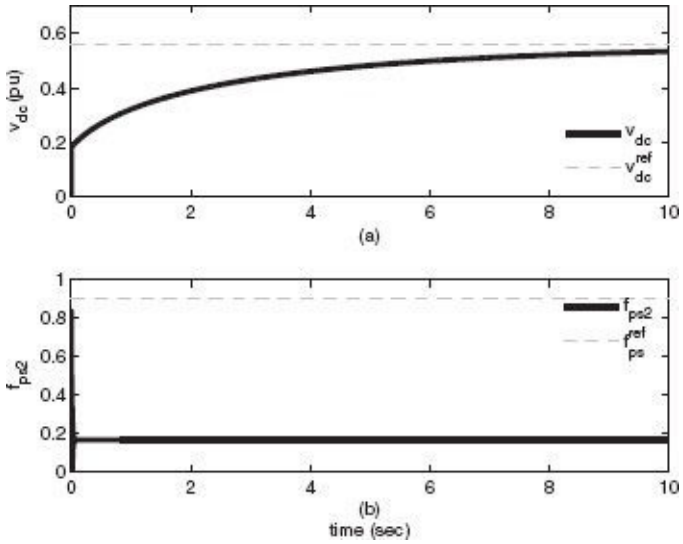

Figure 4.12 System outputs: (a) DC voltage (v_{dc}) and (b) step-up transformer power factor (f_{ps2}).

The DC voltage tracking is presented in Figure 4.12. The DC voltages reach their reference slowly. In Figure 4.13, the control signals v_{dg} and v_{qg} are portrayed, which are bounded and smooth. The DC Link state variables tracking are presented in Figure 4.14; the current (i_{dg}) has a reference to 0 *pu*, because in steady-state, when the DC voltage reference is reached, it is not required to charge the capacitor, $\boldsymbol{i_{ds}^{ss} = 0}$. In order to keep a power factor constant of 0.9, the reactive power (Q_g) should be close to 0 *pu*, which requires $i_{qg} = 0$. The transient process depends on the selection of the P_2 matrix in the inverse optimal control. In Appendix A, a method is proposed to improve the selection of this P_2 matrix.

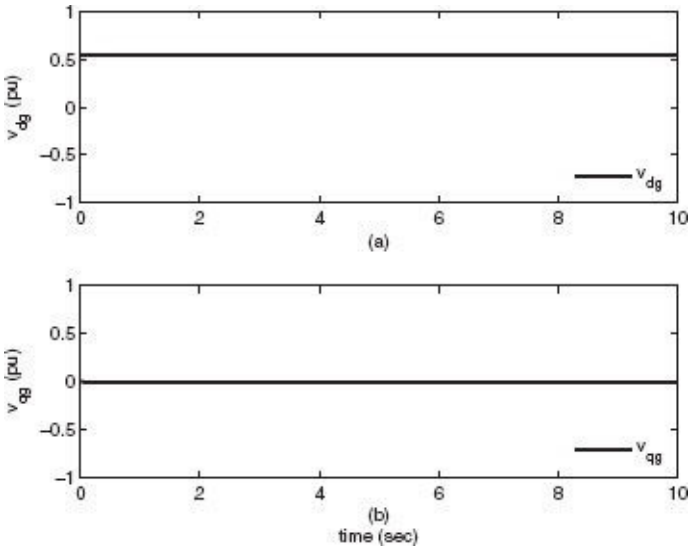

Figure 4.13 Control signals: (a) v_{dg} and (b) v_{qg}.

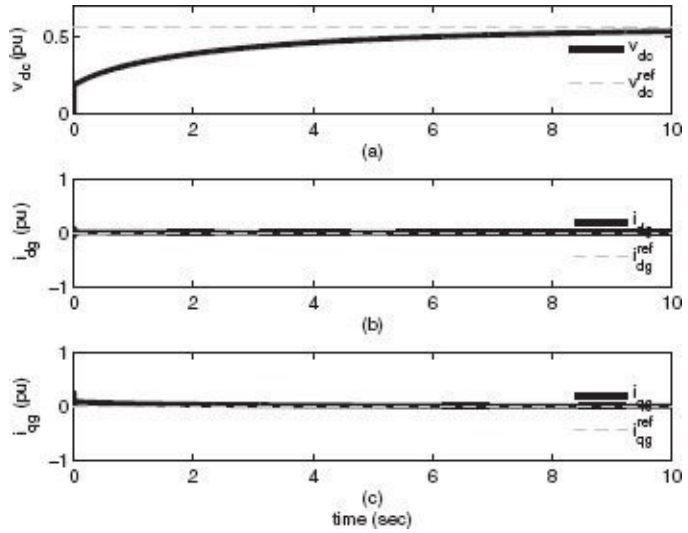

Figure 4.14 State variables: (a) DC voltage v_{dc}, (b) current i_{dg}, and (c) current i_{qg}.

NOTES

[1] MATLAB/Simulink® is a trademark of MathWorks.

[2] MATLAB/Simulink® is a trademark of MathWorks.

5 Neural Network Control of Wind Turbine Induction Generators

All the presented control schemes are based on a doubly fed induction generator (DFIG) mathematical model. The controllers designed in Section 4.1 present an external disturbance robustness; however, the ones developed in Section 4.2 do not have this feature. Additionally, all these controllers are designed with the assumption that all parameters are known. In general, this assumption is not fulfilled in practice. Hence, in order to consider time-varying parameter robustness and external disturbances, in this chapter a recurrent high-order neural network (RHONN) is used to identify the DFIG and DC Link mathematical model; after that, based on this neural model, a block control scheme using sliding modes and the inverse optimal control scheme are utilized to obtain the control law along the same way as in Section 4.1 and 4.2, respectively.

In Section 5.1, the neural identifier structures are described; and the DFIG and DC Link neural identifiers are included in Subsection 5.1.1 and 5.1.2.

In Section 5.2, the sliding modes scheme based on the neural identifiers is presented. In Subsection 5.2.1, the neural block control scheme is used to design the DFIG controller. Additionally, simulation results are presented to validate the control performance. In Subsection 5.2.2, a similar neural network scheme is applied to DC Link and the corresponding simulation results are presented.

In Section 5.3, the inverse optimal control scheme based on the neural mathematical model is presented. In Subsection 5.3.1, the DFIG neural inverse optimal controller is developed; and Subsection 5.3.2 presents the DC Link neural inverse optimal controller. Both of them include the corresponding simulation results.

5.1 NEURAL IDENTIFIERS

In this section, the neural identifiers for the DFIG and DC Link are established. Extended Kalman filter (EKF) is used as the training algorithm for all the neural identifiers.

5.1.1 DFIG NEURAL IDENTIFIER

A neural identifier is used for DFIG modeling; it is assumed that all the state variables are measured. For identification, a RHONN is used as

$$\widehat{\omega}_r\left(k+1\right) = w_{11}\left(k\right) S(\omega_r\left(k\right))^2 + w_{12}\left(k\right) S\left(\omega_r\left(k\right)\right) + w_{13}\tau_e\left(k\right), \tag{5.1}$$

$$\begin{aligned}\hat{i}_{ds}\left(k+1\right) &= w_{21}\left(k\right) S\left(i_{ds}\left(k\right)\right) + w_{22}\left(k\right) S\left(i_{qs}\left(k\right)\right) \\ &\quad + w_{23}\left(k\right) S\left(\omega_r\left(k\right)\right) S\left(i_{ds}\left(k\right)\right) S\left(i_{qs}\left(k\right)\right) + w_{24} v_{dr}\left(k\right),\end{aligned} \tag{5.2}$$

$$\hat{i}_{qs}(k+1) = w_{31}(k) S(i_{qs}(k)) + w_{32}(k) S(i_{ds}(k)) + w_{33}(k) S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) + w_{34} v_{qr}(k), \tag{5.3}$$

$$\hat{i}_{dr}(k+1) = w_{41}(k) S(i_{dr}(k)) + w_{42}(k) S(i_{qr}(k)) + w_{43}(k) S(\omega_r(k)) S(i_{dr}(k)) S(i_{qr}(k)) + w_{44} v_{dr}(k), \tag{5.4}$$

$$\hat{i}_{qr}(k+1) = w_{51}(k) S(i_{qr}(k)) + w_{52}(k) S(i_{dr}) + w_{53}(k) S(\omega_r(k)) S(i_{dr}(k)) S(i_{qr}(k)) + w_{54} v_{qr}(k). \tag{5.5}$$

The training is performed on line, using a series-parallel configuration as displayed in Figure 5.1. All the neural network states are initialized as zero. The covariance matrices are initialized as diagonal, with nonzero elements as $P_1(0) = 920$, $R_1(0) = 900$, $Q_1(0) = 800$ for the rotor speed, and $P_i(0) = 800$, $R_i(0) = 800$, $Q_i(0) = 50$, $i = 2,3,4,5$ for the stator and rotor currents.

The choice of the parameters $P_i(0)$, $Q_i(0)$, $R_i(0)$ is an important step in the design of a neural controller, because it is based on the neural identifier. Then, in Appendix A, the particle swarm optimization (PSO) algorithm is proposed to determine the initial values of these matrices [71].

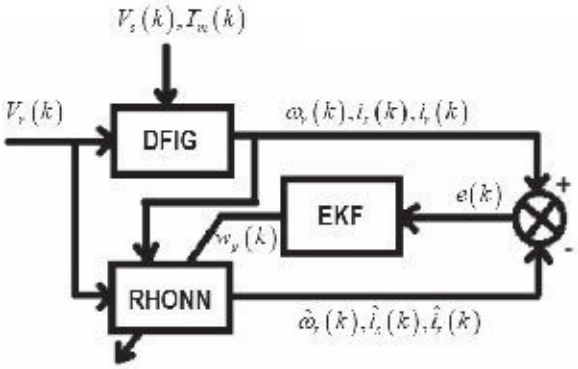

Figure 5.1 Identification scheme for the DFIG.

5.1.2 DC LINK NEURAL IDENTIFIER

A neural identifier is used for DC Link modeling; it is assumed that all the state variables are measured. For identification, a RHONN is defined as

$$\hat{v}_{dc}(k+1) = w_{11} S(v_{dc}(k)) + w_{12} S(v_{dc}(k)) S(i_{qg}(k)) + w_{13} i_{dg}(k), \tag{5.6}$$

$$\hat{i}_{dg}(k+1) = w_{21} S(i_{dg}(k)) + w_{22} S(i_{qg}(k)) + w_{23} S(v_{dc}(k)) + w_{24} v_{dg}(k), \tag{5.7}$$

$$\hat{i}_{qg}(k+1) = w_{31}S(i_{qg}(k)) + w_{32}S(i_{dg}(k)) + w_{33}v_{qg}(k). \tag{5.8}$$

The training is performed on line, using a series-parallel configuration. All the neural network states are initialized as zero. The covariance matrices are initialized as diagonal, with nonzero elements as $P_1(0) = 920$, $R_1(0) = 700$, $Q_1(0) = 100$ for the DC voltage, $P_i(0) = 920$, $R_i(0) = 900$, $Q_i(0) = 100$, $i = 2,3$ for the currents.

Similarly to previous neural identifier, the choice of the initial conditions of $P_i(0)$, $Q_i(0)$, $R_i(0)$ is an important step in the design of a neural controller, because it is based on the neural identifier. Then, in Appendix A, the PSO algorithm is proposed to determine the initial values of these matrices [71].

5.2 NEURAL SLIDING MODES BLOCK CONTROL

In this section, the procedure for design and development of controllers based on the neural identifiers is detailed. The discrete sliding modes algorithm is used to develop the DFIG and the DC Link neural controllers, which are presented in Subsections 5.2.1 and 5.2.2, respectively.

5.2.1 DFIG NEURAL CONTROLLER

The variables to be controlled are the DFIG electric torque ($\tau_e(k)$) and the stator reactive power ($Q_s(k)$). The control objectives are: a) to track a time-varying electric torque reference $\left(\tau_e^{ref}(k)\right)$, and b) to keep the electric power factor ($f_{ps1}(k)$) at the stator terminals constant by means of the stator reactive power control. The control scheme utilized is shown in Figure 5.2.

The electric torque ($\tau_e(k)$) and stator reactive power ($Q_s(k)$) are formulated, respectively, as

$$\tau_e(k) = i_r(k)^T M_{\tau_e} i_s(k) \tag{5.9}$$

and

$$Q_s(k) = v_s(k)^T M_Q i_s(k), \tag{5.10}$$

where $M_{\tau_e} = X_m \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$, $M_Q = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$.

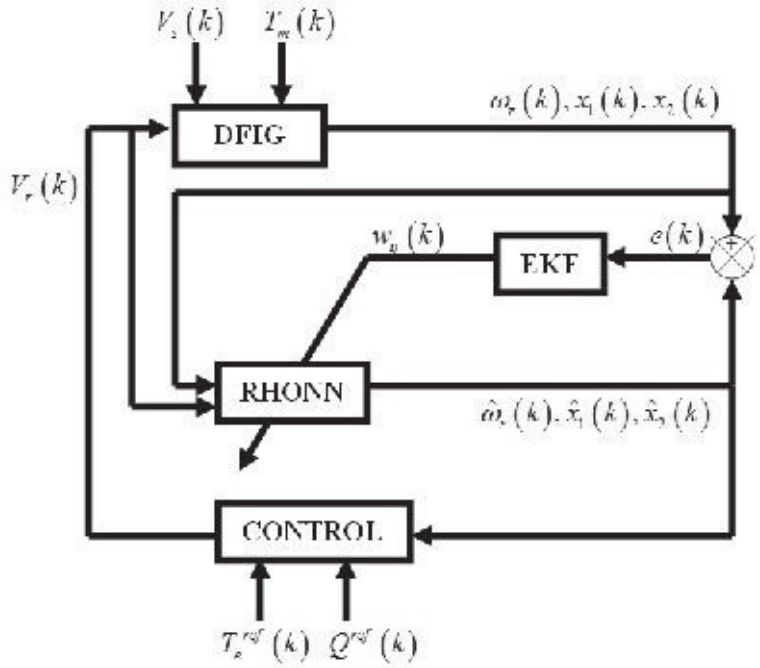

Figure 5.2 Neural control scheme.

The electric torque reference $\left(\tau_e^{ref}\ (k)\right)$ is defined as

$$\tau_e^{ref}\ (k)\ = \gamma_1\ (k)\,, \tag{5.11}$$

where $\gamma_1\ (k)$ is an arbitrary time-varying function, and the stator reactive power reference is defined as a power factor function ($f_{ps1}(k)$) [38, 77]:

$$Q_s^{ref}\ (k)\ = \frac{P_s\ (k)}{f_{ps1}}\sqrt{1-f_{ps1}^2}, \tag{5.12}$$

where $P_s\ (k)\ \approx\ \tau_e^{ref}\ (k)$.

In order to simplify the controller synthesis, the identifier equations are rewritten as

$$\hat{\omega}_r\ (k+1)\ = \hat{f}_{\omega_r}\ (k) + w_{13}\ (k)\,\tau_e\ (k)\,, \tag{5.13}$$

$$\hat{x}_1\ (k+1)\ = \hat{f}_1\ (k) + B'_1 u\ (k)\,, \tag{5.14}$$

$$\hat{x}_2\ (k+1)\ = \hat{f}_2\ (k) + B'_2 u\ (k)\,, \tag{5.15}$$

where

$$\widehat{x}_1(k) = \begin{bmatrix} \hat{i}_{ds}(k) \\ \hat{i}_{qs}(k) \end{bmatrix}, \widehat{x}_2(k) = \begin{bmatrix} \hat{i}_{dr}(k) \\ \hat{i}_{qr}(k) \end{bmatrix},$$

$$\widehat{f}_{\omega_r}(k) = w_{11} S(\omega_r(k))^2 + w_{12} S(\omega(k)),$$

$$\widehat{f}_1(k) = \begin{bmatrix} w_{21} S(i_{ds}(k)) + w_{22} S(i_{qs}(k)) + \\ \dots w_{23} S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \\ w_{31} S(i_{qs}(k)) + w_{32} S(i_{ds}(k)) + \\ \dots w_{33} S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \end{bmatrix}$$

$$\widehat{f}_2(k) = \begin{bmatrix} w_{41} S(i_{dr}(k)) + w_{42} S(i_{qr}(k)) + \\ \dots w_{43} S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \\ w_{31} S(i_{qs}(k)) + w_{32} S(i_{ds}(k)) + \\ \dots w_{33} S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \end{bmatrix}$$

$$B'_1 = \begin{bmatrix} w_{24} & 0 \\ 0 & w_{34} \end{bmatrix}, B'_2 = \begin{bmatrix} w_{44} & 0 \\ 0 & w_{54} \end{bmatrix}.$$

In order to obtain the block system, from (5.9), (5.14), and (5.15) we can define

$$\hat{\tau}_e(k+1) = \widehat{f}_{\tau_e}(k) + B_{\tau_e}(k) u(k), \tag{5.16}$$

where

$$\widehat{f}_{\tau_e}(k) = \widehat{f}_2(k)^T M_{\tau_e} \widehat{f}_1(k) + u(k)^T B'^T_2 M_{\tau_e} B'_1 u(k),$$
$$B_{T_e}(k) = \widehat{f}_1(k)^T M^T_{\tau_e} B'_2 + \widehat{f}_2(k)' M_{\tau_e} B'_1.$$

For control synthesis the input weights are assumed fixed and equal. Then, B'_1 and B'_2 are constant with equal entries; with these conditions, the non linear term $u(k)^T B'^T_2 M_{\tau_e} B'_1 u(k) = 0$ in $\widehat{f}_{\tau_e}(k)$, then $\widehat{f}_{\tau_e}(k) = \widehat{f}_2(k)^T M_{\tau_e} \widehat{f}_1(k)$. From (5.10), (5.14), and (5.15), we obtain the stator reactive power equation in differences as

$$\widehat{Q}_s(k+1) = \widehat{f}_{Q_s}(k) + B_{Q_s}(k) u(k), \tag{5.17}$$

where

$$\widehat{f}_{Q_s}(k) = v_s(k)^T M_{Q_s} \widehat{f}_1(k),$$
$$B_{Q_s}(k) = v_s(k)^T M_{Q_s} B'_1(k).$$

From (5.16) and (5.17), the block system is defined as

$$x_1(k+1) = f_{x_1}(k) + g_{x_1}(k)\,u(k), \tag{5.18}$$

where

$$x_1(k) = \begin{bmatrix} \tau_e(k) \\ Q_s(k) \end{bmatrix}, \; f_{x_1}(k) = \begin{bmatrix} \widehat{f}_{\tau_e}(k) \\ \widehat{f}_{Q_s}(k) \end{bmatrix},$$
$$g_{x_1}(k) = \begin{bmatrix} B_{\tau_e}(k) \\ B_{Q_s}(k) \end{bmatrix}.$$

Clearly, (5.18) constitutes a one block system [49]; then it can be used to synthetize the control input $u(k)$ as stated in the following theorem.

Theorem 5.1

For system (5.18), the control law defined as

$$u^c(k) = u^{equ}(k) + u^{din}(k)$$

ensures trajectory tracking, with $u^{equ}(k)$ determined using discrete time sliding modes [80] and $u^{din}(k)$ a proportional term.

Proof. Using the discrete time sliding modes, the sliding manifold is defined as

$$s(k) = x_1(k) - x_1^{ref}(k), \tag{5.19}$$

where $x_1^{ref}(k) = \begin{bmatrix} \tau_e^{ref}(k) \\ Q_s^{ref}(k) \end{bmatrix}$.

Evaluating (5.19) at $(k + 1)$, the equivalent control $u^{equ}(k)$ is calculated as [80]

$$u^{equ}(k) = -g_{x_1}(k)^{-1}\left(f_{x_1}(k) - x_1^{ref}(k+1)\right). \tag{5.20}$$

Applying $u(k) = u^{equ}(k)$ to (5.18), the state of the closed-loop system reaches the sliding manifold $s(k) = 0$ in one sampling period. However, it is appropriate to add to the control signal a stabilizing term $u^{equ}(k)$, in order to reach the sliding surface asymptotically and to avoid high gain control, defined as

$$u^{din}(k) = -g_{x_1}(k)^{-1}(Ks(k)), \tag{5.21}$$

where K is a Schur matrix [81]. Applying $u(k)$ in (5.18), the system dynamics of (5.19) are

obtained as

$$s(k+1) = Ks(k). \tag{5.22}$$

Then the sliding manifold $s(k) = 0$ is reached asymptotically.

To take into account the boundedness of the control signal $\|u(k)\| < u_{\max}$, $u_{\max} > 0$, where $\|\cdot\|$ stands for the Euclidean norm, the following control law is selected [80]:

$$u(k) = \begin{cases} u_{\max} \frac{u^c(k)}{\|u^c(k)\|} & if \ \|u^c(k)\| > u_{\max} \\ u^c(k) & if \ \|u^c(k)\| \leq u_{\max} \end{cases}. \tag{5.23}$$

The stability proof using (5.23) is presented in [80].

5.2.1.1 Simulation Results

To evaluate the performance of the proposed controller, a simulation for a three-phase generator with a stator-referred rotor is developed. The generator parameters appear in Table 3.1.

The simulation conditions are:

Simulation time: 12.5 seconds.

Sampling time: $t_s = 0.5\ ms$.

DFIG initial conditions: rotor speed 0.3 *pu*, $i_{ds} = 0.001\ pu$, $i_{qs} = 0.001\ pu$, $i_{dr} = 0.001\ pu$, $i_{qr} = 0.001\ pu$.

Identification input is a chirp signal, frequency range 0-60*Hz* and amplitude 0.1 *pu*.

The 2.5-second initials are the identification; after that, the control signal is incepted at 2.5 seconds.

The initial electric torque reference is a constant signal at 0.4 *pu*.

From 1 to 3 seconds a pulse variation in the electric torque reference with amplitude of 0.5 *pu* is incepted.

At 5 seconds the electric torque reference is changed to a senoidal signal centered at 0.5 *pu* with amplitude of 0.4 *pu* and 1 *Hz*.

Power factor reference is constant in 0.9.

The gain K in (5.21) is defined as $\begin{bmatrix} 0.95 & 0 \\ 0 & 0 \end{bmatrix}$.

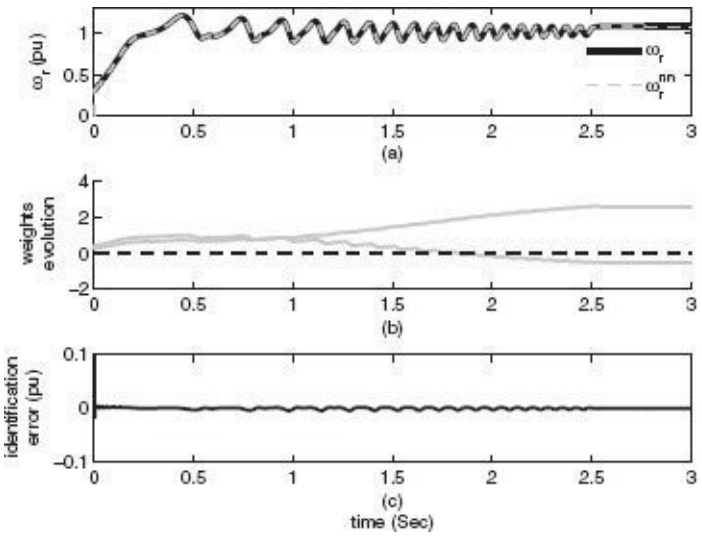

Figure 5.3 (a) Rotor speed (ω_r) with rotor speed identifier, (b) neural network weights evolution, and (c) identification error, respectively.

The behavior of the neural identifier is shown in Figure 5.3 to Figure 5.7. In these figures, the DFIG variables are presented jointly with their identifiers. It can be seen that all the identification errors are small; additionally, all the neural network weights are bounded as shown in part (b) of Figure 5.3 to Figure 5.7. At 2.5 seconds the control signals are incepted. Figure 5.8(a) presents the electric torque τ_e. In this figure, it can be seen that the tracking for electric torque is reached quickly. In Figure 5.8(b), the reactive power is presented. The reference trackings are ensured by the neural controller, despite even if the references are time-varying signals. In Figure 5.8(c), the power factor is displayed and it can be seen that the tracking is good enough in presence of electric torque reference variations. The DFIG current performances are shown in Figure 5.9, where we can see that the DFIG currents are within nominal limits. The control signals are bounded as shown in Figure 5.10. In Figure 5.11, the rotor speed is presented, which has small variations due to the electric torque tracking error.

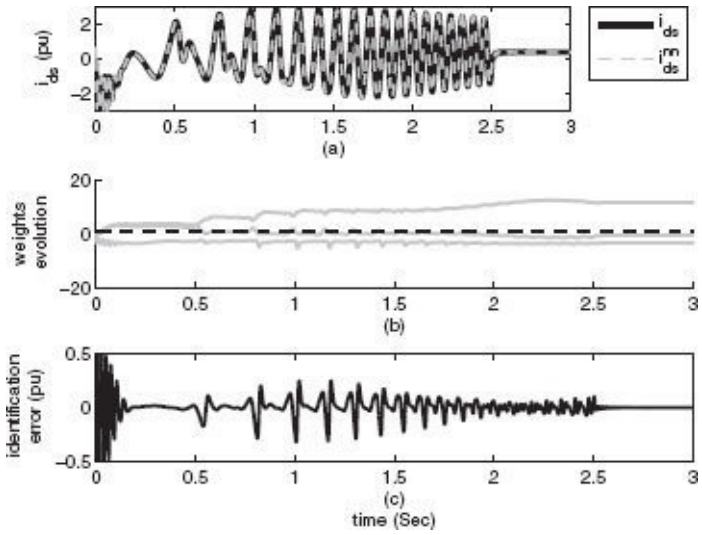

Figure 5.4 (a) Stator current (i_{ds}) with stator current identifier, (b) neural network weights evolution, and (c) identification error, respectively.

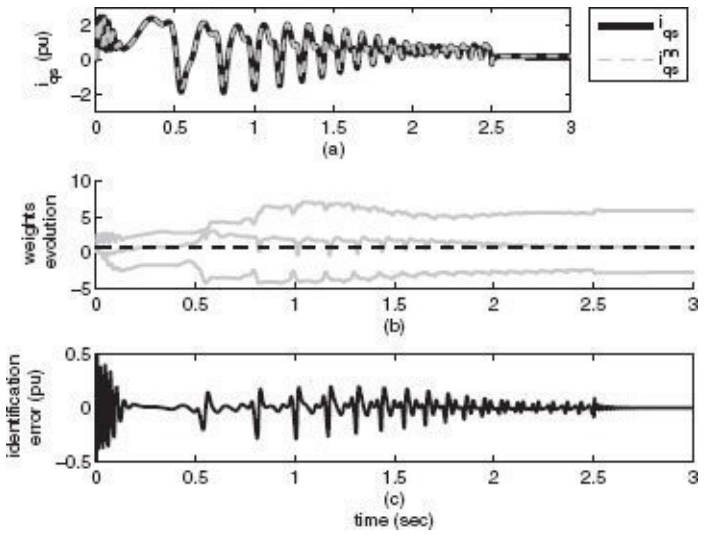

Figure 5.5 (a) Stator current (i_{qs}) with stator current identifier, (b) neural network weights evolution, and (c) identification error, respectively.

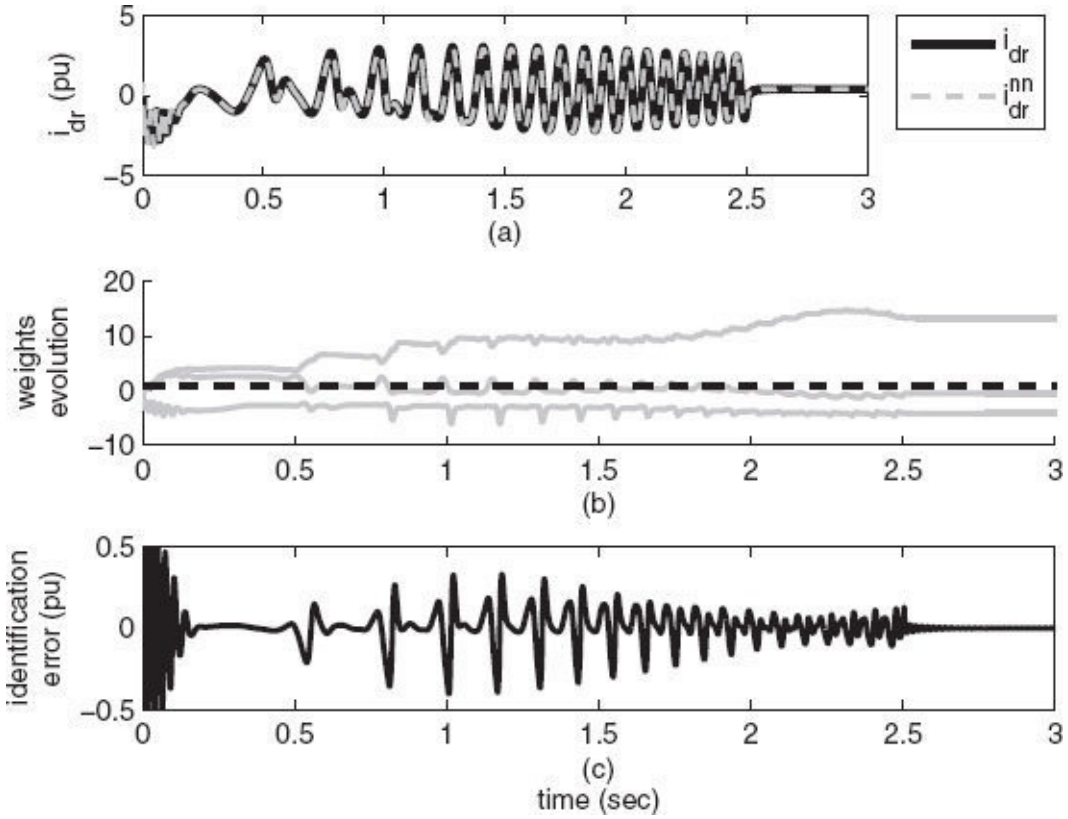

Figure 5.6 (a) Rotor current (i_{dr}) with rotor current identifier, (b) neural network weights evolution, and (c) identification error, respectively.

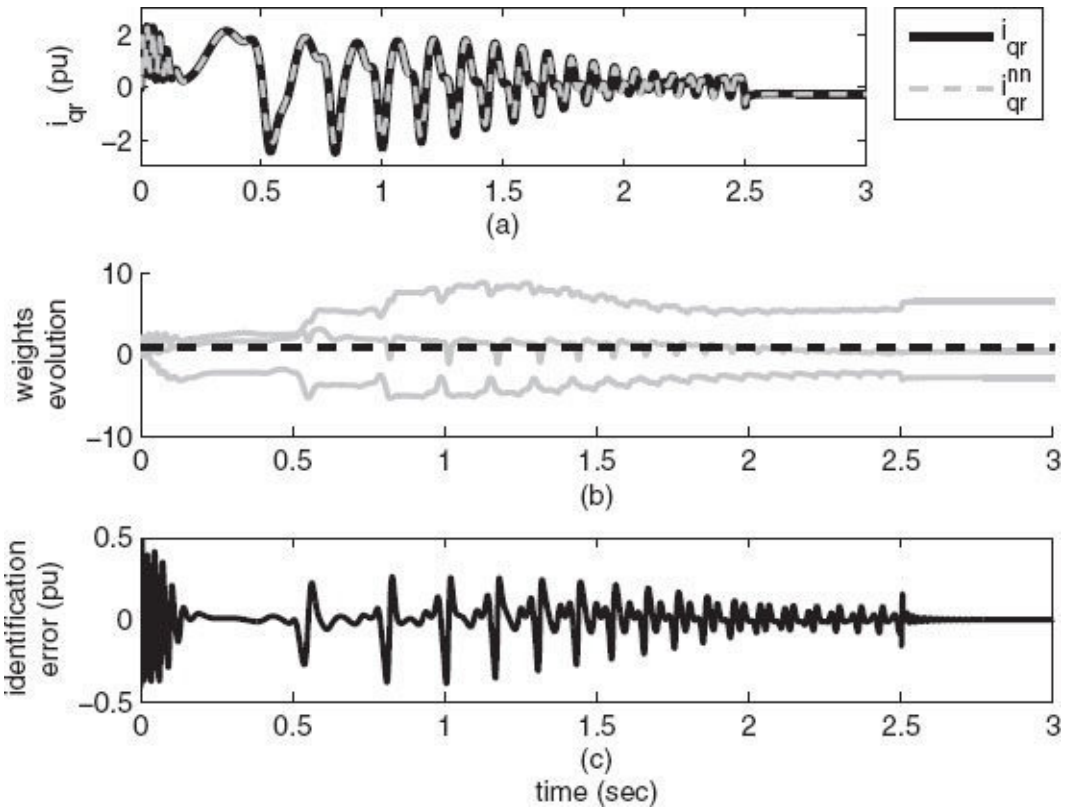

Figure 5.7 (a) Rotor current (i_{qr}) with rotor current identifier, (b) neural network weights evolution, and (c) identification error, respectively.

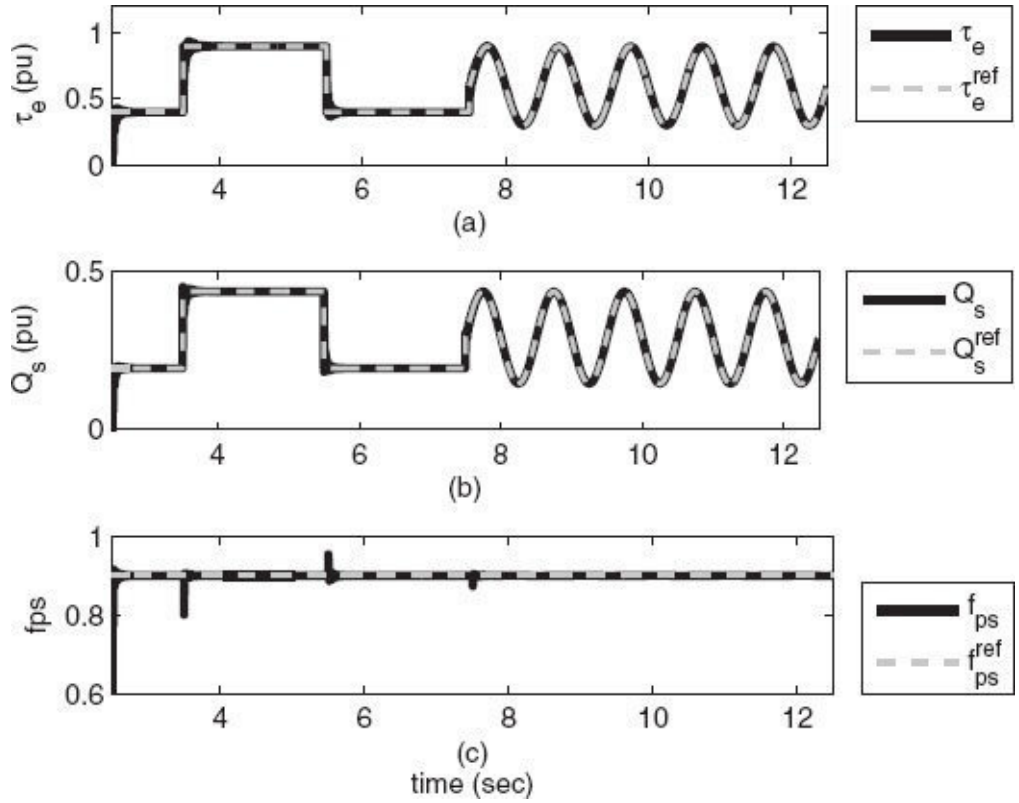

Figure 5.8 System outputs: (a) electric torque (τ_e) tracking, (b) reactive power (Q_s) tracking, and (c) power factor (f_{ps1}) tracking.

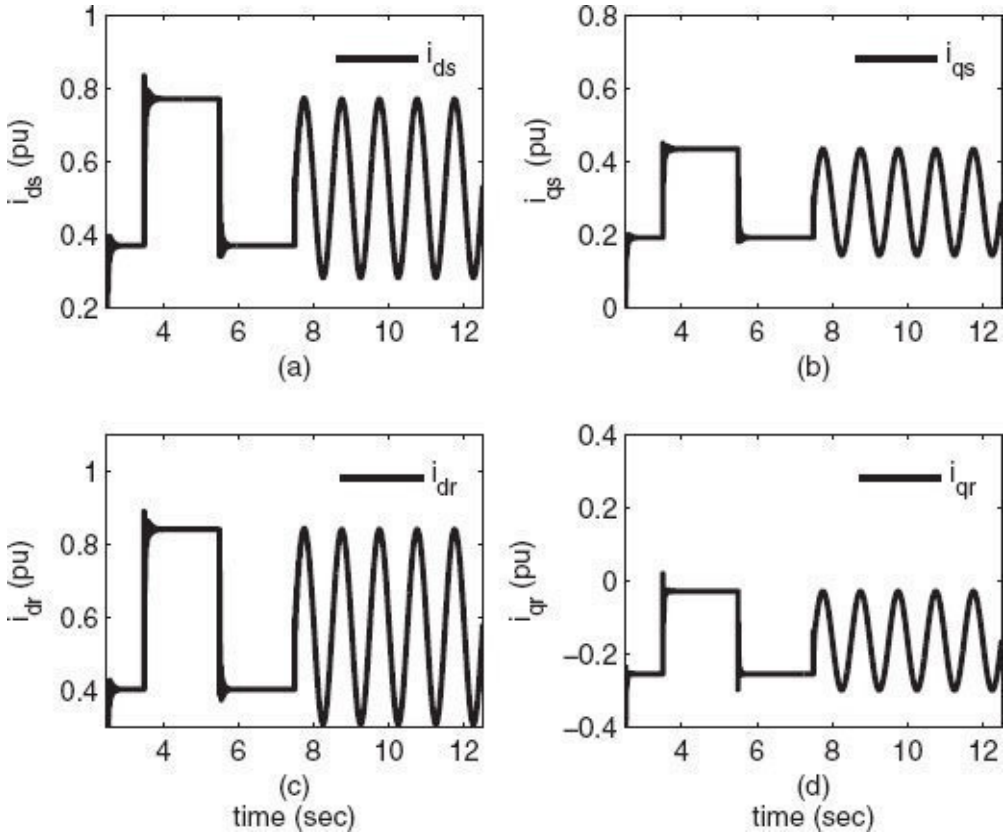

Figure 5.9 Generator currents: (a) stator current i_{ds}, (b) stator current i_{qs}, (c) rotor current i_{dr}, and (b) rotor current i_{qr}.

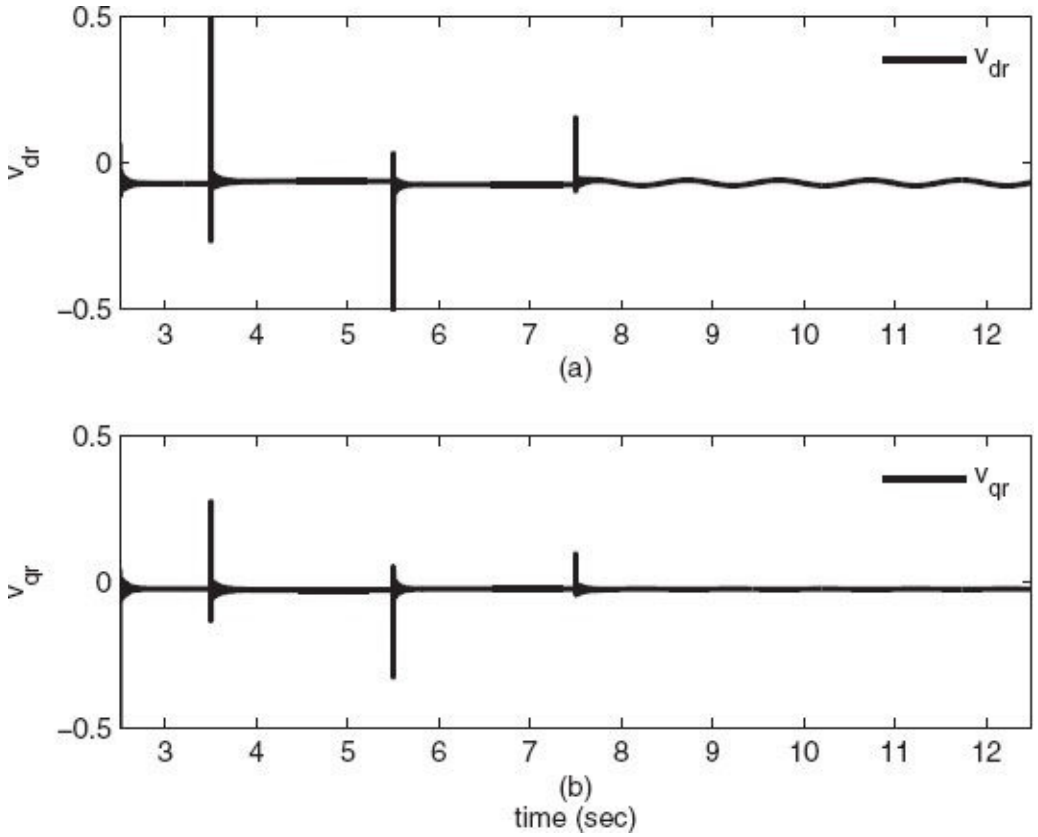

Figure 5.10 Control signals: (a) v_{dr} and (b) v_{qr}.

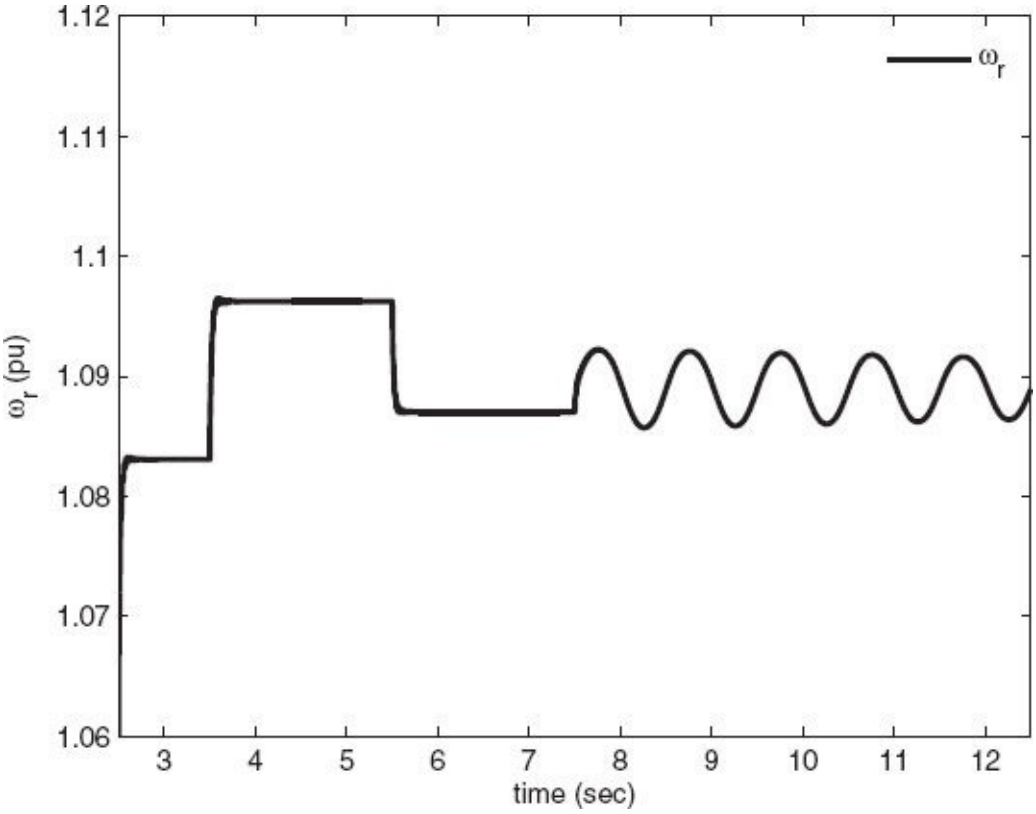

Figure 5.11 Rotor speed (ω_r).

5.2.2 DC LINK NEURAL CONTROLLER

The variables to be controlled are the capacitor voltage ($v_{dc}(k)$) and the reactive power ($Q_g(k)$). The control objectives are: a) to track a DC voltage reference $\left(v_{dc}^{ref}(k)\right)$ on the DC Link, and b) to keep constant the electric power factor ($f_{ps2}(k)$) at the stepup transformer terminals by means of the reactive power ($Q_g(k)$) control.

The step-up transformer reactive power ($Q_g(k)$) is formulated as

$$Q_g(k) = v_{sg}(k)^T M_Q i_g(k). \tag{5.24}$$

The DC voltage reference $\left(v_{dc}^{ref}(k)\right)$ is defined as

$$v_{dc}^{ref}(k) = \gamma_2(k), \tag{5.25}$$

where $\gamma_2(k)$ is an arbitrary time-varying function, and the reference for the reactive power is defined as a function of the electric power factor (f_{ps2}) [77]:

$$Q_g^{ref}(k) = \frac{P_g(k)}{f_{ps2}}\sqrt{1-f_{ps2}^2}, \tag{5.26}$$

where $P_g(k) = v_{sg}(k)^T M_P i_g(k)$, and $\boldsymbol{M_P} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Let us define the tracking error for the DC voltage as

$$\varepsilon_1^g(k) = v_{dc}(k) - v_{dc}^{ref}(k). \tag{5.27}$$

From (5.27), using (5.6), then $\boldsymbol{\varepsilon_1^g(k+1)}$ is equal to

$$\varepsilon_1^g(k+1) = w_{11} S(v_{dc}(k)) + w_{12} S(v_{dc}(k)) S(i_{qg}(k)) + w_{13} i_{dg}(k) - v_{dc}^{ref}(k+1), \tag{5.28}$$

where it can be seen that the DC voltage ($v_{dc}(k)$) is controlled directly by $i_{dg}(k)$.

Then, the i_{dg} reference is defined as

$$i_{dg}^{ref}(k) = w_{13}^{-1}\left(v_{dc}^{ref}(k+1) + k_1\varepsilon_1^g(k) - w_{11} S(v_{dc}(k)) - w_{12} S(v_{dc}(k)) S(i_{qg}(k))\right), \tag{5.29}$$

where $\boldsymbol{k_1\varepsilon_1^g(k)}$ is introduced to reach the reference asymptotically, with $|k_1| < 1$. On the other hand, the tracking error for the reactive power is

$$\varepsilon_2^g(k) = Q_g(k) - Q_g^{ref}(k). \tag{5.30}$$

From (5.24), and considering that $v_{qgs}(k) = 0$, it could be established that

$$Q_g(k) = -v_{dgs}(k) i_{qg}(k). \tag{5.31}$$

In order to determine the reference $\left(i_{qg}^{ref}\right)$, we assume that $\boldsymbol{\varepsilon_2^g(k) = 0}$, then $\boldsymbol{Q_g(k) = Q_g^{ref}(k)}$; therefore, it is easy to see that $\boldsymbol{i_{qg}^{ref}}$ is given by

$$i_{qg}^{ref}(k) = -i_{dg}^{ref}(k)\frac{\sqrt{1 - f_{ps2}^2}}{f_{ps2}}. \tag{5.32}$$

Considering $\boldsymbol{x_2(k)} = \begin{bmatrix} \hat{i}_{dg}(k) \\ \hat{i}_{qg}(k) \end{bmatrix}$, equations (5.7) and (5.8) can be rewritten as follows:

$$x_2(k+1) = f_{x_2}(k) + g_{x_2} u_g(k) \tag{5.33}$$

with

$$f_{x_2}(k) = \begin{bmatrix} w_{21} S(i_{dg}(k)) + w_{22} S(i_{qg}(k)) + w_{23} S(v_{dc}(k)) \\ w_{31} S(i_{qg}(k)) + w_{32} S(i_{dg}(k)) \end{bmatrix},$$

$$g_{x_2} = \begin{bmatrix} w_{24} & 0 \\ 0 & w_{33} \end{bmatrix}.$$

Then, it is clear that equation (5.33) is of the form (5.18); according to Theorem 5.1, the control input $u_g(k)$ is selected as follows.

At first, the sliding manifold is formulated as

$$s_g(k) = x_2(k) - x_2^{ref}(k), \tag{5.34}$$

where $x_2^{ref}(k) = \begin{bmatrix} i_{dg}^{ref}(k) \\ i_{qg}^{ref}(k) \end{bmatrix}$.

Evaluating (5.34) at $(k+1)$ and using (5.33),

$$s_g(k+1) = f_{x_2}(k) + g_{x_2} u_g(k) - x_2^{ref}(k+1). \tag{5.35}$$

Then, the equivalent control $u_g^{equ}(k)$ is calculated as [80]

$$u_g^{equ}(k) = -g_{x_2}^{-1}\left(f_{x_2}(k) - x_2^{ref}(k+1)\right). \tag{5.36}$$

Applying $u(k) = u_g^{equ}(k)$ to (3.10), the state of the closed-loop system reaches the sliding manifold $sg(k) = 0$ in one sample time. However, it is appropriate to add to the control signal a stabilizing term $u_g^{din}(k)$ in order to reach the sliding surface asymptotically and to avoid high gain control; hence, the complete control $u_g^c(k)$ is proposed as

$$u_g^c(k) = u_g^{equ}(k) + u_g^{din}(k), \tag{5.37}$$

where

$$u_g^{din}(k) = g_{x_2}^{-1}(K_g s_g(k)) \tag{5.38}$$

and $K_g = \begin{bmatrix} k_1^g & 0 \\ 0 & k_2^g \end{bmatrix}$ is a Schur matrix [81]. To take into account the boundedness of the control signal $\|u_g(k)\| < u_{g\max}, u_{g\max} > 0$, where $\|\cdot\|$ stands for the Euclidean norm, the following control law is selected [80]:

$$u_g(k) = \begin{cases} u_{g\max} \frac{u_g^c(k)}{\|u_g^c(k)\|} & if \ \|u_g^c(k)\| > u_{g\max} \\ u_g^c(k) & if \ \|u_g^c(k)\| \leq u_{g\max} \end{cases}. \tag{5.39}$$

The stability proof using (5.39) is presented in [80].

5.2.2.1 Simulation Results

To evaluate the performance of the proposed controller a simulation for a DC Link is developed. The DC Link parameters appear in Table 3.1. The simulation is performed in MATLAB/Simulink®[1], with conditions:

Simulation time: 10 seconds.

Sampling time: $t_s = 0.5\ ms$.

DC Link initial conditions: $v_{dc} = 0.01\ pu$, $i_{dg} = 0\ pu$, $i_{qg} = 0\ pu$.

Identification input is a chirp signal, frequency range 0-60Hz, and amplitude 0.01 pu.

The first 1 second is the identification; after that, the control signal is incepted at 1 second.

The DC voltage reference is a constant signal at 0.5567 pu.

Power factor reference is constant at 0.9 pu.

Load resistance R_L of 383.0579 pu connected to the capacitor in parallel scheme in order to simulate an unknown perturbation.

The behavior of the neural identifier is shown in Figure 5.12 to Figure 5.14. In these figures, the DC Link variables are presented jointly with their identifiers; it can be seen that all the identification errors are small; additionally, all the neural network weights are bounded as shown in part (b) of Figure 5.12 to Figure 5.14. At 1 second the control signal is incepted.

The performance of the DC Link controller is shown in Figure 5.15 to Figure 5.17. The DC voltage (Figure 5.15(a)) and power factor tracking (Figure 5.15(b)) is controlled to the reference. In this figure, we can see that the DC voltage reaches the reference quickly, and the electric power factor is kept constant at 0.9 during the 10-second lapse. The control signals v_{dg} and v_{qg} are bounded as shown in Figure 5.16. The state variables of the DC Link (3.9), and (3.10) are shown in Figure 5.17, where it can be seen that the state variables are stable and bounded. The transient lapse is short and the resistance R_L is

connected in parallel scheme to simulate the capacitor discharge by a resistive load.

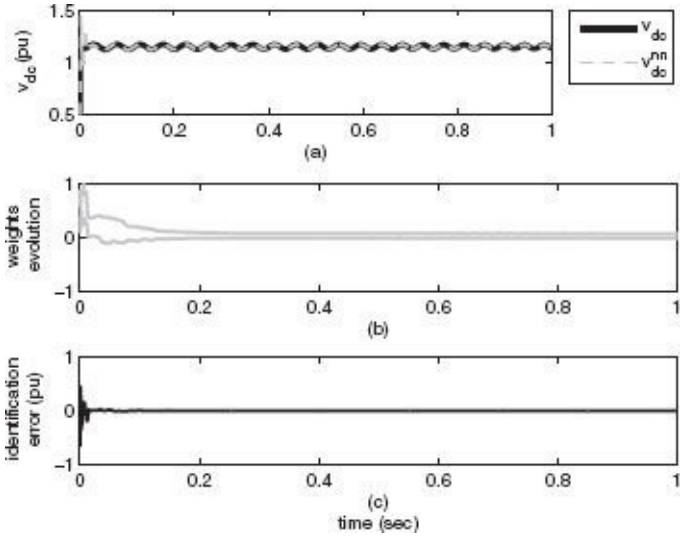

Figure 5.12 (a) DC voltage (v_{dc}) with the respective identifier, (b) neural network weights evolution, and (c) identification error.

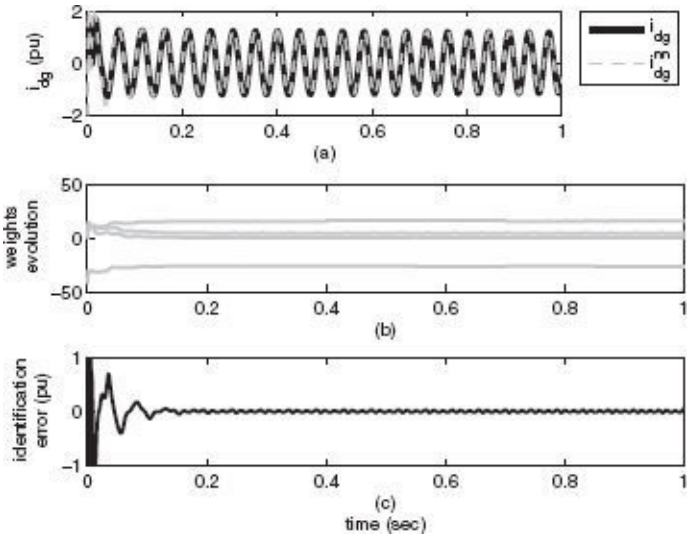

Figure 5.13 DC Link current (i_{dg}) with the respective identifier, (b) neural network weights evolution, and (c) identification error.

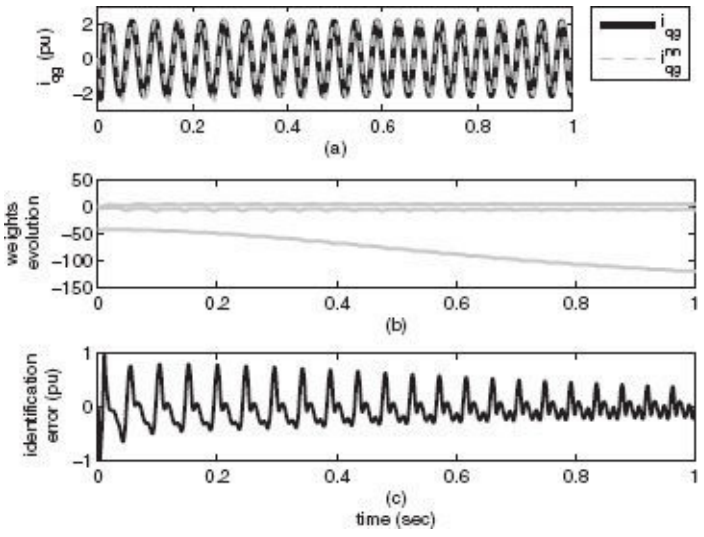

Figure 5.14 DC Link current (i_{qg}) with the respective identifier, (b) neural network

weights evolution, and (c) identification error.

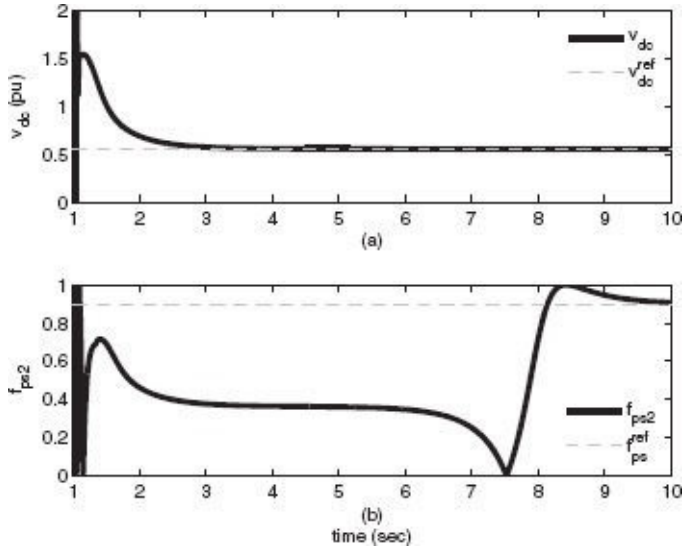

Figure 5.15 System outputs: (a) DC voltage (v_{dc}) and (b) step-up transformer power factor (f_{ps2}).

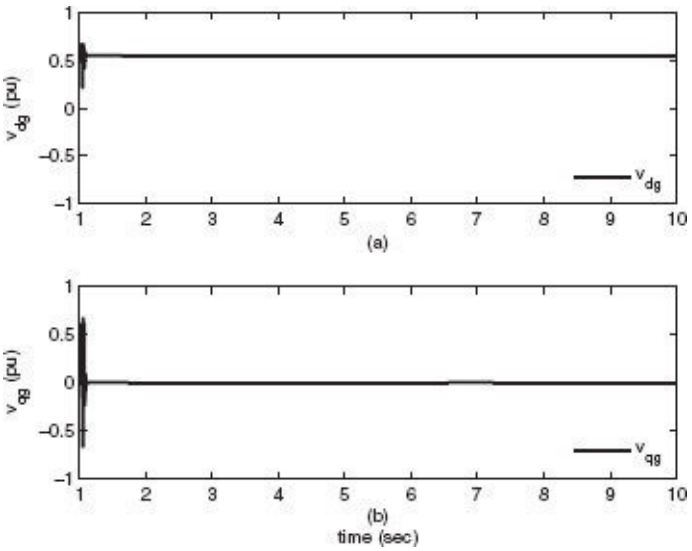

Figure 5.16 Control signals: (a) v_{dg} and (b) v_{qg}.

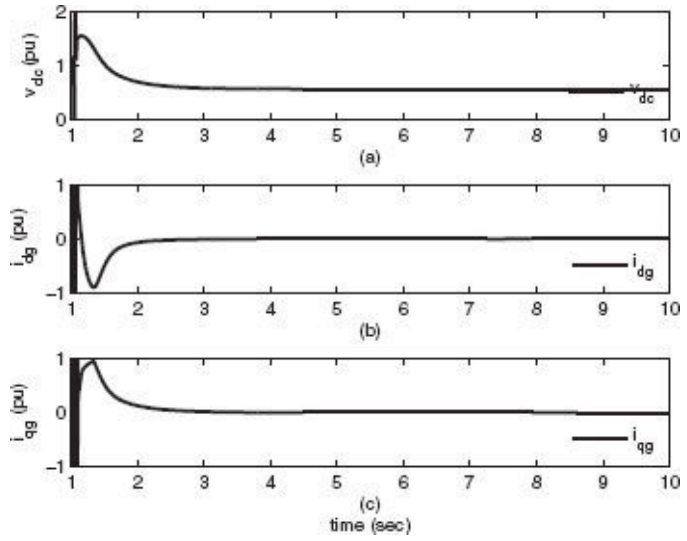

Figure 5.17 State variables: (a) DC voltage v_{dc}, (b) current i_{dg}, and (c) current i_{qg}.

5.3 NEURAL INVERSE OPTIMAL CONTROL

In this section, the inverse optimal control is based on the neural identifiers for the DFIG and DC Link controller development.

5.3.1 DFIG NEURAL CONTROLLER

The variables to be controlled are the DFIG electric torque ($\tau_e(k)$) and the stator reactive power ($Q_s(k)$). The control objectives are: a) to track an electric torque trajectory $\left(\tau_e^{ref}(k)\right)$, and b) to keep the electric power factor ($f_{ps1}(k)$) at the stator terminals constant by means of the stator reactive power control. The electric torque ($\tau_e(k)$) and stator reactive power ($Q_s(k)$) are defined, respectively, as (5.9) and (5.10).

In order to apply the inverse optimal control, we need to calculate the reference for the system state, which is obtained by considering the system steady state as the desired references, and then achieving trajectory tracking as a stabilization problem; hence the system steady stable is obtained as follows. The electric torque reference $\left(\tau_e^{ref}(k)\right)$ is defined as

$$\tau_e^{ref}(k) = \gamma_1(k), \tag{5.40}$$

where $\gamma_1(k)$ is an arbitrary time-varying function, and the reactive power reference is defined as a function of electric power factor (f_{ps1}) [38, 77]:

$$Q_s^{ref}(k) = \frac{P_s(k)}{f_{ps1}}\sqrt{1-f_{ps1}^2},\ P_s(k) \approx \tau_e^{ref}(k). \tag{5.41}$$

The tracking errors are defined, respectively, as

$$\varepsilon_{\tau_e}(k) = \tau_e(k) - \tau_e^{ref}(k), \tag{5.42}$$

$$\varepsilon_{Q_s}(k) = Q_s(k) - Q_s^{ref}(k). \tag{5.43}$$

In order to calculate the steady state, it is assumed that $\varepsilon_{\tau_e}(k) = 0$ and $\varepsilon_{Q_s}(k) = 0$; then using (5.9) and (5.10), equations (5.42) and (5.43) can be rewritten as

$$i_r^{ss}(k)^T M_{\tau_e} i_s^{ss}(k) = \tau_e^{ref}(k), \tag{5.44}$$

$$v_s(k)^T M_{Q_s} i_s^{ss}(k) = Q_s^{ref}(k), \tag{5.45}$$

where the superscript *ss* denotes steady state. From (5.46) and (5.47),

$$i_s(k+1) = i_s(k) + t_s\left(A_{11}(k)\, i_s(k) + A_{12}(k)\, i_r(k)\right) + t_s\left(D_1 v_s(k) + B_1 u(k)\right), \tag{5.46}$$

$$i_r(k+1) = i_r(k) + t_s\left(A_{21}(k)\, i_s(k) + A_{22}(k)\, i_r(k)\right) + t_s\left(D_2 v_s(k) + B_2 u(k)\right), \tag{5.47}$$

the relation between stator and rotor currents in steady state is obtained as

$$i_r^{ss}(k) = G_1 i_s^{ss}(k) + H_1 v_s(k), \tag{5.48}$$

$$i_s^{ss}(k) = G_2 i_r^{ss}(k) + H_2 v_s(k), \tag{5.49}$$

where G_1, H_1, G_2, and H_2 are defined in [69].

Solving i_{ds}^{ss}, i_{qs}^{ss}, i_{dr}^{ss}, and i_{qr}^{ss} of (5.44), (5.45), (5.48), and (5.49), the $i^{ss}(k)$ is defined as

$$i^{ss}(k) = \left[\, i_{ds}^{ss}(k) \;\; i_{qs}^{ss}(k) \;\; i_{dr}^{ss}(k) \;\; i_{qr}^{ss}(k) \,\right]^T. \tag{5.50}$$

Now, based on the DFIG neural identifier, the inverse optimal controller is developed. In order to simplify the controller synthesis, the identifier equations can be rewritten as:

$$\hat{\omega}_r(k+1) = f_{\omega_r}(k) + w_{13}(k)\, T_e(k), \tag{5.51}$$

$$\hat{i}_s(k+1) = f_1(k) + B'_1(k)\, u(k), \tag{5.52}$$

$$\hat{i}_r(k+1) = f_2(k) + B'_2(k)\, u(k), \tag{5.53}$$

where

$$\hat{i}_s(k) = \begin{bmatrix} \hat{i}_{ds}(k) \\ \hat{i}_{qs}(k) \end{bmatrix}, \hat{i}_r(k) = \begin{bmatrix} \hat{i}_{dr}(k) \\ \hat{i}_{qr}(k) \end{bmatrix},$$

$$f_{\omega_r}(k) = w_{11} S(\omega_r(k))^2 + w_{12} S(\omega(k)),$$

$$f_1(k) = \begin{bmatrix} w_{21} S(i_{ds}(k)) + w_{22} S(i_{qs}(k)) + \\ \dots w_{23} S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \\ w_{31} S(i_{qs}(k)) + w_{32} S(i_{ds}(k)) + \\ \dots w_{33} S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \end{bmatrix},$$

$$f_2(k) = \begin{bmatrix} w_{41} S(i_{dr}(k)) + w_{42} S(i_{qr}(k)) + \\ \dots w_{43} S(w_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \\ w_{31} S(i_{qs}(k)) + w_{32} S(i_{ds}(k)) + \\ \dots w_{33} S(\omega_r(k)) S(i_{ds}(k)) S(i_{qs}(k)) \end{bmatrix},$$

$$B'_1(k) = \begin{bmatrix} w_{24}(k) & 0 \\ 0 & w_{34}(k) \end{bmatrix},$$

$$B'_2(k) = \begin{bmatrix} w_{44}(k) & 0 \\ 0 & w_{54}(k) \end{bmatrix}.$$

The system denoted by (5.52) and (5.53) is rewritten as

$$\hat{i}(k+1) = \widehat{f}_{rsc}(k) + \hat{g}_{rsc}(k) u(k), \tag{5.54}$$

where

$$\widehat{f}_{rsc}(k) = \begin{bmatrix} f_1(k) \\ f_2(k) \end{bmatrix}, \hat{g}_{rsc}(k) = \begin{bmatrix} B'_1(k) \\ B'_2(k) \end{bmatrix}.$$

In order to apply the inverse optimal control, the tracking error is defined as

$$i_e(k) = \hat{i}(k) - i^{ss}(k). \tag{5.55}$$

Evaluating (5.55) at time $(k+1)$ and using (5.54), the error dynamic is given by

$$\begin{aligned} i_e(k+1) &= \hat{i}(k+1) - i^{ss}(k+1) \\ i_e(k+1) &= \widehat{f}_{rsc}(k) + \hat{g}_{rsc}(k) u(k) - i^{ss}(k+1). \end{aligned} \tag{5.56}$$

For system (5.56), the control signal ($u(k)$) is decomposed into two components:

$$u(k) = u_1(k) + u_i^*(k), \tag{5.57}$$

and in order to convert (5.56) in to the form (2.35), $u_1(k)$ is selected as

$$u_1(k) = \hat{g}_{rsc}(k)^{*-1}\left(i^{ss}(k+1)\right),$$

where $\hat{g}_{rsc}(k)^{*-1} = \left(\hat{g}_{rsc}(k)^T \hat{g}_{rsc}(k)\right)^{-1} \hat{g}_{rsc}(k)^T$. Then, system (5.56), with (5.57) as input, results in

$$i_e(k+1) = \hat{f}_{rsc}(k) + \hat{g}_{rsc}(k) u_i^*(k), \tag{5.58}$$

which has the form of (2.35) as

$$x(k+1) = f_i(x(k)) + g_i(x(k)) u_i^*(k),$$

where $x(k) := i_e(k)$ is the system state, $f_i(x(k)) = \hat{f}_{rsc}(k)$, $g_i(x(k)) = \hat{g}_{rsc}(k)$, and hence the inverse optimal control law $\left(u_i^*(k)\right)$ is established using Theorem 2.1 as

$$\begin{aligned} u_i^*(k) = & -\tfrac{1}{2}\left(R_1(x(k)) + \tfrac{1}{2} g_i^T(x(k)) P_1\right. \\ & \left. \times g_i(x(k))\right)^{-1} g_i^T(x(k)) P_1 f_i(x(k)), \end{aligned} \tag{5.59}$$

where for the rotor side converter (RSC) controller, $R_1(x(k))$ and P_1 are the $R(x(k))$ and P matrices in (2.45), respectively.

5.3.1.1 Simulation Results

To evaluate the performance of the proposed controller, a simulation for a three-phase generator with a stator-referred rotor is developed. The generator parameters appear in Table 3.1.

The simulation conditions are:

Simulation time: 12.5 seconds.

Sampling time: t_s 0.5 *ms*.

DFIG initial conditions: rotor speed 0.3 *pu*, i_{ds} = 0.001 *pu*, i_{qs} = 0.001 *pu*, i_{dr} = 0.001 *pu*, i_{qr} = 0.001 *pu*.

Identification input is a chirp signal, frequency range 0-60 *Hz*, and amplitude 0.1 *pu*.

The 2.5-second initial is the identification; after that, the control signal is incepted at 2.5 seconds.

The initial electric torque reference is a constant signal at 0.4 *pu*.

From 1 to 3 seconds a pulse variation in the electric torque reference with amplitude of 0.5 *pu* is incepted.

At 5 seconds the electric torque reference is changed to a senoidal signal centered at 0.5 *pu* with amplitude of 0.4 *pu* and 1 *Hz*.

Power factor reference is constant at 0.9.

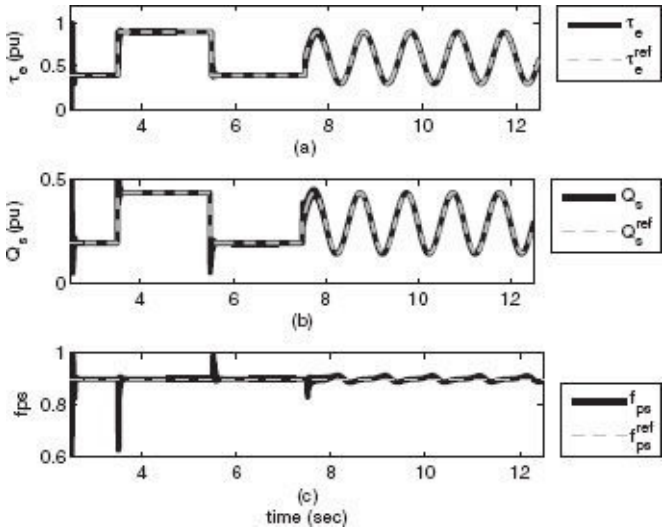

Figure 5.18 System outputs: (a) electric torque (τ_e) tracking, (b) reactive power (Q_s) tracking, and (c) power factor (f_{ps1}) tracking.

The behavior of the neural identifier is shown in Figure 5.3 to Figure 5.7. The first 2.5 seconds the identification is achieved; after that, the control signals are incepted. Figure 5.18 presents the electric torque (τ_e) tracking (a), the reactive power (Q_s) tracking (b), and electric power factor (c). In this figure, it can be seen that the tracking for electric torque and the reactive power are reached fast. The reference trackings are ensured by the neural control algorithm despite that the references are time-varying signals. The power factor is kept constant when the electric torque reference is a pulse signal; and the power factor has a small variation when the electric torque reference is a senoidal signal. The DFIG current performances are shown in Figure 5.19, where we can see that the DFIG currents reach their respective references. The control signals are bounded and these are shown in Figure 5.20. Rotor speeds associated with this experiment are shown in Figure 5.21.

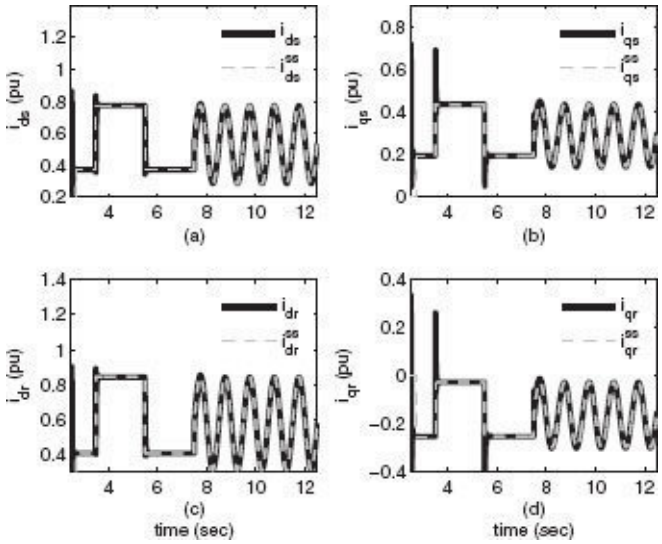

Figure 5.19 Generator currents: (a) stator current i_{ds}, (b) stator current i_{qs}, (c) rotor current i_{dr}, and (b) rotor current i_{qr}.

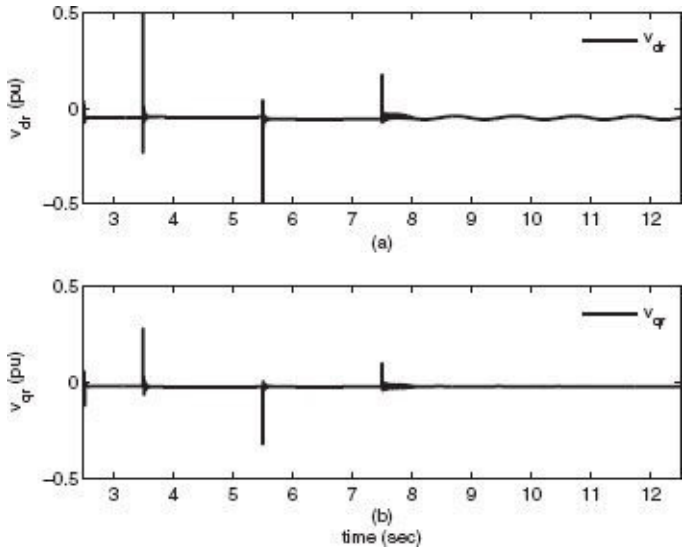

Figure 5.20 Control signals: (a) v_{dr} and (b) v_{qr}.

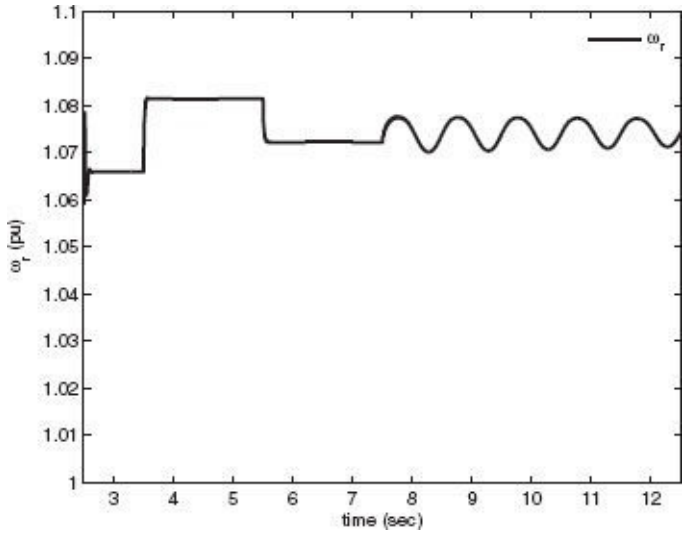

Figure 5.21 Rotor speed (ω_r).

5.3.2 DC LINK NEURAL CONTROLLER

The variables to be controlled are the capacitor voltage ($v_{dc}(k)$) and the reactive power ($Q_g(k)$). The control objectives are: a) to track a DC voltage reference $\left(v_{dc}^{ref}(k)\right)$ for the DC Link, and b) to keep the electric power factor ($f_{ps2}(k)$) constant at the step-up transformer terminals by means of the reactive power control ($Q_g(k)$).

The step-up transformer reactive power ($Q_g(k)$) is defined as

$$Q_g(k) = v_{sg}(k)^T M_Q i_g(k). \tag{5.60}$$

The DC voltage reference $\left(v_{dc}^{ref}(k)\right)$ is defined as

$$v_{dc}^{ref}(k) = \gamma_2(k), \tag{5.61}$$

where $\gamma_2(k)$ is an arbitrary time-varying function, and the reference for the reactive power is defined as a function of electric power factor (f_{ps2}) [77]:

$$Q_g^{ref}(k) = \frac{P_g(k)}{f_{ps2}}\sqrt{1-f_{ps2}^2},\ P_g(k) = v_{sg}(k)^T M_P i_g(k). \tag{5.62}$$

Similar to the previous case (DFIG controller), it is required to determine the steady-state values for the v_{dc}^{ref} and Q_g^{ref}. The tracking error for the DC voltage and the reactive power are defined, respectively, as

$$\varepsilon_{v_{dc}}(k) = v_{dc}(k) - v_{dc}^{ref}(k) \tag{5.63}$$

$$\varepsilon_{Q_g}(k) = Q_g(k) - Q_g^{ref}(k) \tag{5.64}$$

Evaluating (5.63) at time ($k+1$) and using (5.65),

$$v_{dc}(k+1) = v_{dc}(k) + t_s\left(\frac{1}{Cv_{dc}(k)}v_{gs}^T(k) M_P i_g(k)\right) \tag{5.65}$$

yields

$$\varepsilon_{v_{dc}}(k+1) = v_{dc}(k) + \frac{t_s}{Cv_{dc}(k)}v_{gs}^T(k) M_P i_g(k) - v_{dc}^{ref}(k+1). \tag{5.66}$$

Assuming that in steady state $\varepsilon_{v_{dc}}(k) = \varepsilon_{v_{dc}}(k+1) = \varepsilon_{Q_g}(k) = \varepsilon_{Q_g}(k+1) = 0$, then $v_{dc}^{ss}(k) = v_{dc}^{ref}(k)$. Then solving to i_{dg}^{ss}, i_{qg}^{ss} of (5.64) and (5.66), new variables $\hat{x}_g(k) = \left[\hat{v}_{dc}(k)\ \hat{i}_{dg}(k)\ \hat{i}_{qg}(k)\right]^T$, $x_g^{ss}(k) = \left[v_{dc}^{ss}(k)\ i_{dg}^{ss}(k)\ i_{qg}^{ss}(k)\right]^T$ are defined, and the tracking error is written as

$$\varepsilon_{x_g}(k) = \hat{x}_g(k) - x_g^{ss}(k). \tag{5.67}$$

In order to simplify the controller synthesis, the DC Link identifier equations can be rewritten as

$$\hat{x}_g(k+1) = \hat{f}_{x_g}(k) + \hat{g}_{x_g}(k)\,u(k), \tag{5.68}$$

where

$$\hat{f}_{x_g}(k) = \begin{bmatrix} w_{11}S(v_{dc}) + w_{12}S(v_{dc})\,S(i_{qg}) + w_{13}i_{dg} \\ w_{21}S(i_{dg}) + w_{22}S(i_{qg}) + w_{23}S(v_{dc}) \\ w_{31}S(i_{qg}) + w_{32}S(i_{dg}) \end{bmatrix},$$

$$\hat{g}_{x_g}(k) = \begin{bmatrix} 0 & 0 \\ w_{24}(k) & 0 \\ 0 & w_{33}(k) \end{bmatrix}.$$

Evaluating (5.67) at time $(k+1)$, the tracking error dynamics are obtained as

$$\varepsilon_{x_g}(k+1) = \hat{f}_{x_g}(k) + \hat{g}_{x_g}(k)\,u(k) - x_g^{ss}(k+1). \tag{5.69}$$

Then the control signal $u_g(k)$ is decomposed in two components as

$$u_g(k) = u_2(k) + u_g^*(k), \tag{5.70}$$

where

$$u_2(k) = \hat{g}_{x_g}(k)^{*-1}\left(x_g^{ss}(k+1)\right), \tag{5.71}$$

with $\hat{g}_{x_g}(k)^{*-1} = \left(\hat{g}_{x_g}(k)^T\,\hat{g}_{x_g}(k)\right)^{-1}\hat{g}_{x_g}(k)^T$.

Then system (5.69), with (5.70) as input, results in

$$\varepsilon_{x_g}(k+1) = \hat{f}_{x_g}(k) + \hat{g}_{x_g}(k)\, u_g^*(k). \tag{5.72}$$

System (5.72) is of the form (2.35), hence the proposed inverse optimal control law $u_g^*(k)$ using Theorem 2.1 becomes

$$\begin{aligned} u_g^*(k) = & -\frac{1}{2}\left(R_2(x(k)) + \frac{1}{2}\hat{g}_{x_g}(k)^T P_2 \hat{g}_{x_g}(k)\right)^{-1} \\ & \times \hat{g}_{x_g}(k)^T P_2 \hat{f}_{x_g}(k), \end{aligned} \tag{5.73}$$

where for the grid side converter (GSC) controller, $R_2(x(k))$ and P_2 are the $R(x(k))$ and P matrices in (2.45), respectively.

5.3.2.1 Simulation Results

To evaluate the performance of the proposed controller a simulation for a DC Link is developed. The DC Link parameters appear in Table 3.1. The simulation is performed in MATLAB/Simulink®[2]

Simulation time: 7 seconds.

Sampling time: $t_s = 0.5\ ms$.

DC Link initial conditions: $v_{dc} = 0.01\ pu$, $i_{dg} = 0\ pu$, $i_{qg} = 0\ pu$.

Identification input is a chirp signal, frequency range 0-60 Hz and amplitude 0.01 pu.

The first 1 second is the identification; after that, the control signal is incepted at 1 second.

The DC voltage reference is a constant signal at 0.5567 pu.

Power factor reference is constant at 0.9.

Load resistance R_L of 383.0579 pu connected to the capacitor in parallel scheme in order to simulate an unknown perturbation.

The behavior of the neural identifier is not presented in this section; it is shown in Figure 5.12 to Figure 5.14 in Section 5.2.2. At 1 second the control signals is incepted. The performance of the DC Link controller is shown in Figure 5.22 to Figure 5.24. The DC voltage (Figure 5.22(a)) and power factor tracking (Figure 5.22(b)) are controlled to the reference. In this figure, we can see that the DC voltage reaches the reference quickly, and the electric power factor is kept constant in 0.9, with some numerical discontinuities due to the electric power factor calculated as $f_{ps2} = \frac{P_g}{\sqrt{P_g^2+Q_g^2}}$. Then the discontinuities occur when P_g crosses zero and $Q_g \approx 0$. The control signals v_{dg} and v_{qg} are bounded as shown in Figure 5.23. The state variables of the DC Link (3.9), and (3.10) are shown in Figure 5.24, where we can see that the state variables are stable and bounded. The

transient time is short and the resistance R_L is connected in parallel scheme to simulate the capacitor discharge by a resistive load.

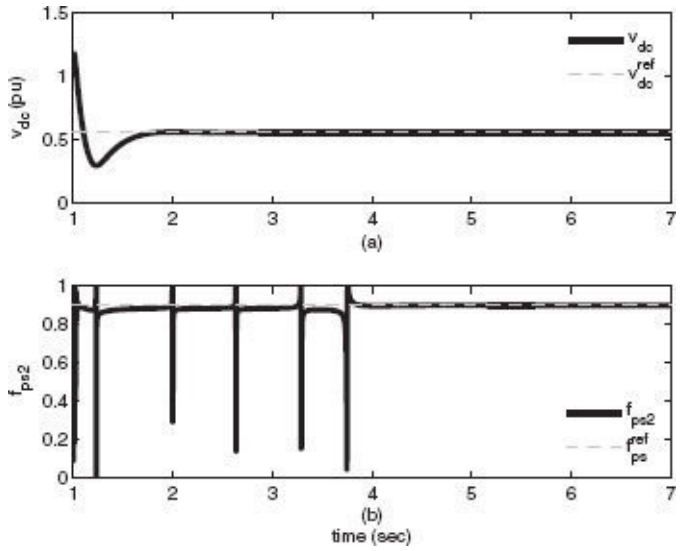

Figure 5.22 System outputs: (a) DC voltage (v_{dc}) and (b) step-up transformer power factor (f_{ps2}).

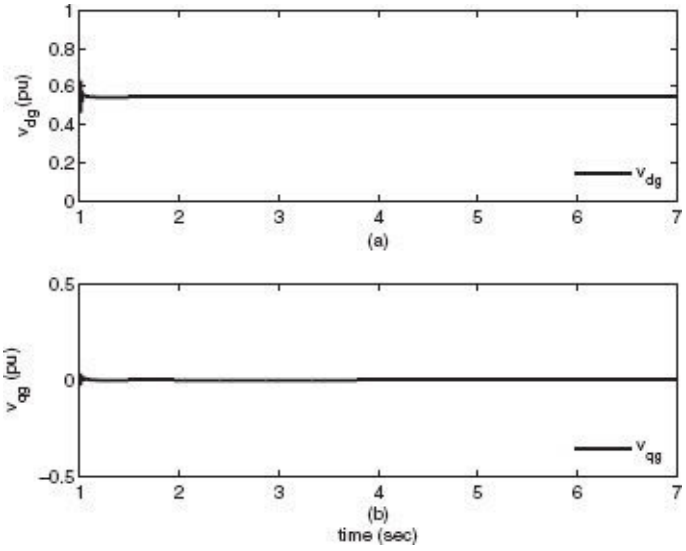

Figure 5.23 Control signals: (a) v_{dg} and (b) v_{qg}.

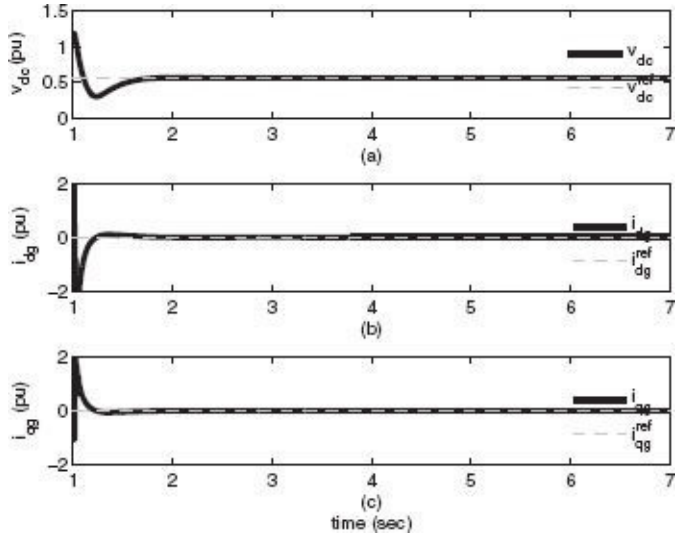

Figure 5.24 State variables: (a) DC voltage v_{dc}, (b) current i_{dg}, and (c) current i_{qg}.

NOTE

[1] MATLAB/Simulink® is a trademark of MathWorks., with conditions:

[2] MATLAB/Simulink® is a trademark of MathWorks., with conditions:

6 Implementation of Wind Energy Testbed

So far, the algorithms developed throughout this book have proven their performance by means simulations. The real time implementation of control algorithms represents a great challenge because during the controller design, hypotheses are considered to facilitate the development and often in a implementation. Frequently in real time implementations, all hypotheses cannot be fulfilled. There are also physical limitations on the prototype that affects the implementation.

In this chapter the real time implementation of the designed algorithms is presented. Additionally, a comparison of the statistic information of the results is also included.

6.1 REAL-TIME CONTROLLER PROGRAMING

The dSpace[1] DS1104 signal acquisition board provides libraries that are compatible with the software MATLAB/Simulink®[2]. Additionally, the dSpace company provides a monitoring software named *ControlDesk*, which allows to monitor and interact in real time with the control algorithm loaded in the data acquisition board.

If the *ControlDesk* software is installed correctly, the DS1104 card libraries are also automatically loaded when the MATLAB is loaded, too. It is verified in the command window of MATLAB, as shown in Figure 6.1.

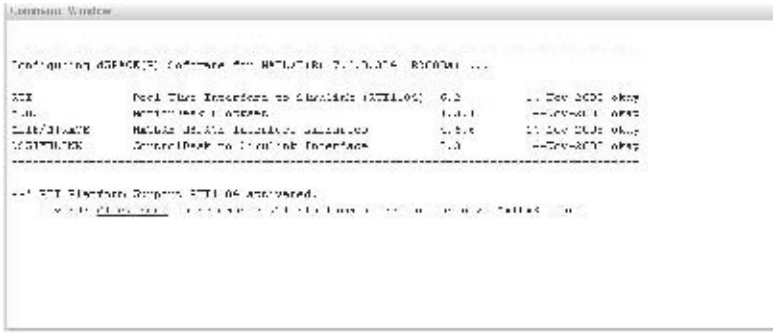

Figure 6.1 Command window of MATLAB.

The *ControlDesk* software adds new blocks to Simulink, which are useful for the design of an algorithm using the data acquisition board hardware. The new blocks can be used as a standard block, which facilitates the incorporation of these libraries into a Simulink model, as shown in Figure 6.2. The dSPACE RTI1104 libraries can be used to access the analog-to-digital converters (ADC), as well as pulse width modulation (PWM) output ports.

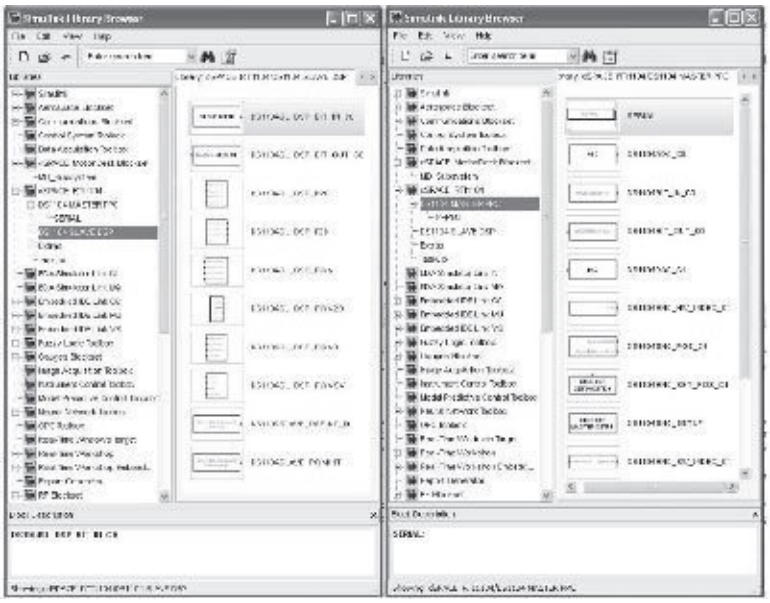

Figure 6.2 DS1104 Simulink libraries.

The control algorithms previously designed in this book are implemented in a Simulink model, where the input signals are the ADC ports and the output signals are the PWM ports, as shown in Figure 6.3. Once the Simulink model is implemented and after quick configurations, the algorithm compilation is done automatically by pressing the button *Incremental Build* in the toolbar.

When the Simulink model is compiled, a code file with extension *.sdf* is generated. It is loaded directly to the DS1104 board, which executes the code in real time. The *ControlDesk* allows the monitoring and interaction of the variables within the algorithm in real time. The interaction of the user with the algorithm is performed by fully customizable interface, which can be designed using the virtual instruments available in the *ControlDesk*. An interface to monitor the algorithm previously designed can be seen in Figure 6.4. The main advantage of using a data acquisition board DS1104 of dSpace is that it simplifies the real time implementation of an algorithm programmed in Simulink.

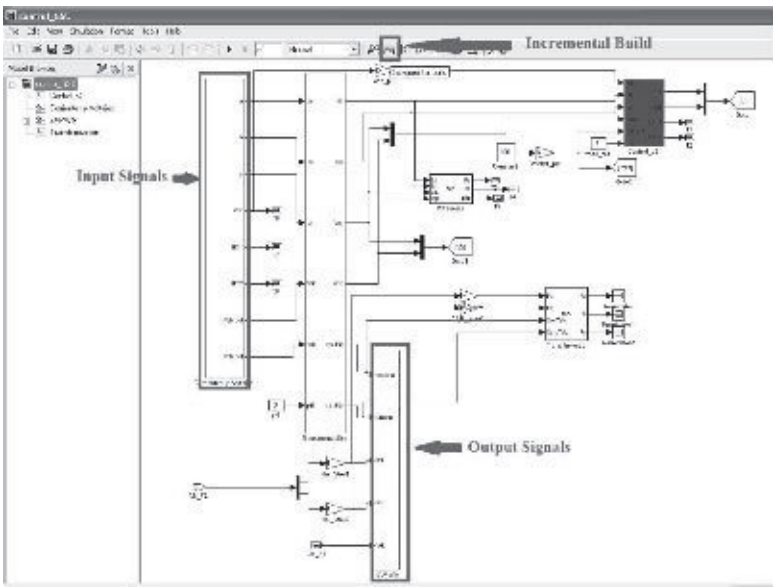

Figure 6.3 Simulink model.

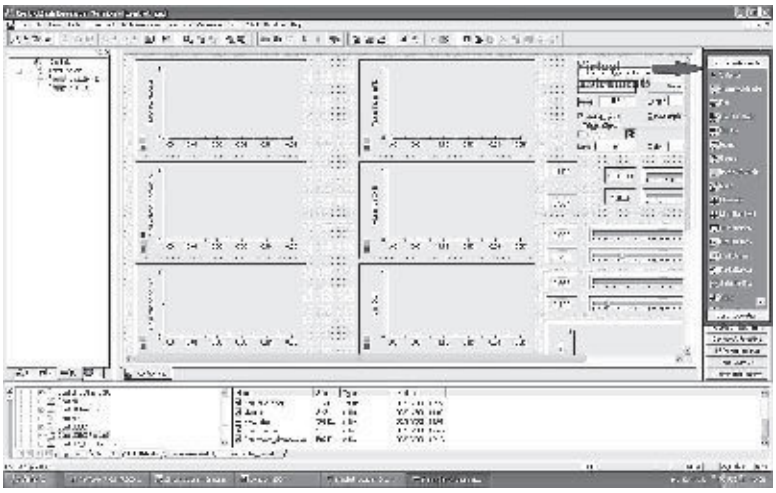

Figure 6.4 *ControlDesk* interface.

6.2 DOUBLY FED INDUCTION GENERATOR PROTOTYPE

In order to evaluate the performance of the proposed controller schemes, a low power (1/4 HP) prototype is integrated. The doubly fed induction generator (DFIG) prototype complete is displayed in Figure 6.5, for which the nominal parameters appear in Table 6.1. This prototype includes four major parts: a 1/4 HP three-phase DFIG, a DC motor, two PWM units for the power stage, and a personal computer (PC) for supervising, which has the data acquisition board installed.

Table 6.1
Parameters of Doubly Fed Induction Generator Prototype

Symbol	Parameter	Value
X_m	Magnetizing Reactance	2.3175 *pu*
X_s	Stator Reactance	2.4308 *pu*
X_r	Rotor Reactance	2.4308 *pu*
r_s	Stator Windings Resistance	0.1609 *pu*
r_r	Rotor Windings Resistance	0.0502 *pu*
H	Angular Moment of Inertia	0.23 *sec*
ω_b	Base Angular Frequency	376.99112 *rad/sec*
P_b	Base Power	185.4 *VA*
V_b	Base Voltage	179.63 *V*
X_l	Three Phase Lines Reactance	0.0045 *pu*
r_g	Three Phase Lines Resistance	0.0014 *pu*
C	DC Link Capacitance	0.1854 *pu*

The scheme and corresponding pictures of the prototype are included as follows. Figure 6.6 shows a schematic representation of the prototype, used for experiments. Just the DFIG is presented in Figure 6.7, which was acquired with Labvolt[3]. Figure 6.8 presents the DC motor (Baldor[4] 3/4 HP), which is used to emulate a wind turbine, coupled to the DFIG. Figure 6.9 shows a view of the PC and the DS1104[5] data acquisition board, which allows to download applications directly from Simulink. Figure 6.10 portrays the encoder coupled between the DFIG and the DC motor to sense the mechanical rotor speed, and Figure 6.11 shows one of the PWM drivers. The connection of the DFIG prototype to the power system is done through a three-phase auto-transformer, which is displayed in Figure 6.12.

The control scheme implementation are performed using MATLAB/Simulink®[6] with a DS1104 data acquisition board. The presence of unmodeled dynamics is one of the main challenges for a real time implementation.

Table 6.2 presents a resume of the control schemes implemented successfully in the DFIG prototype. In Section 6.3, 6.4, and 6.5, the sliding modes control, neural sliding modes control, and neural inverse optimal control performance are presented, respectively. Each section presents the DFIG outputs and the DC Link outputs controlled for the control scheme proposed and their respective control signals as well. It is important that all controllers are tested under the same real time conditions for the same output objectives. For all the implementations, the DC motor imposes a constant rotor speed in 0.97 *pu* as we can see in Figure 6.13.

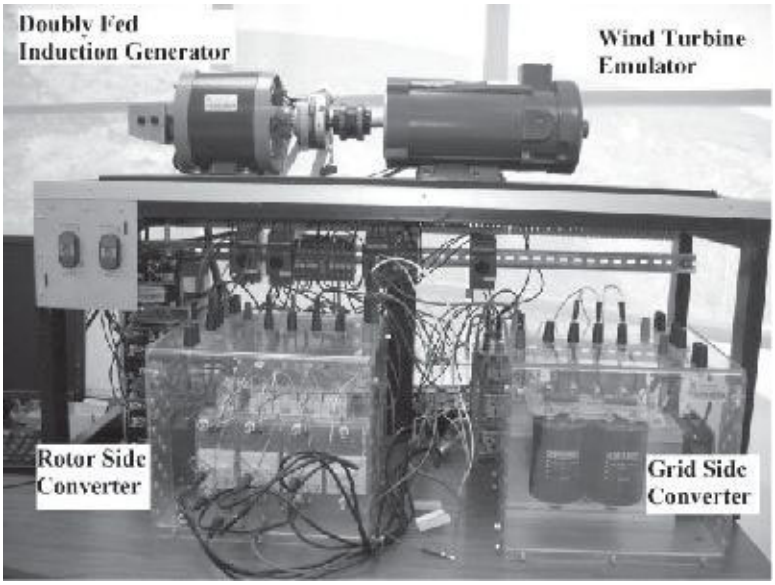

Figure 6.5 DFIG prototype description.

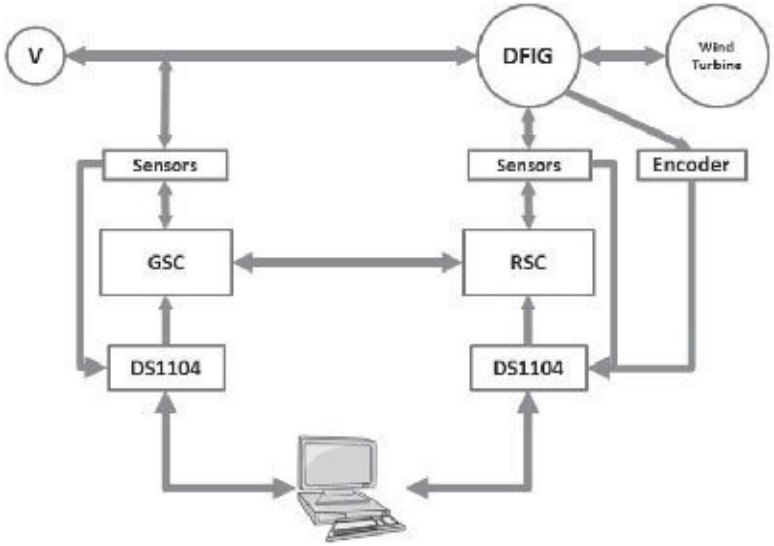

Figure 6.6 Prototype scheme.

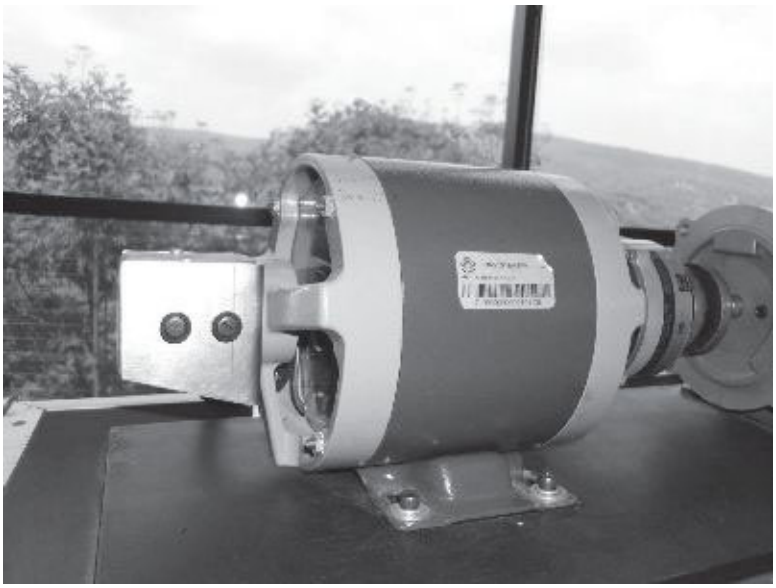

Figure 6.7 1/4 HP DFIG.

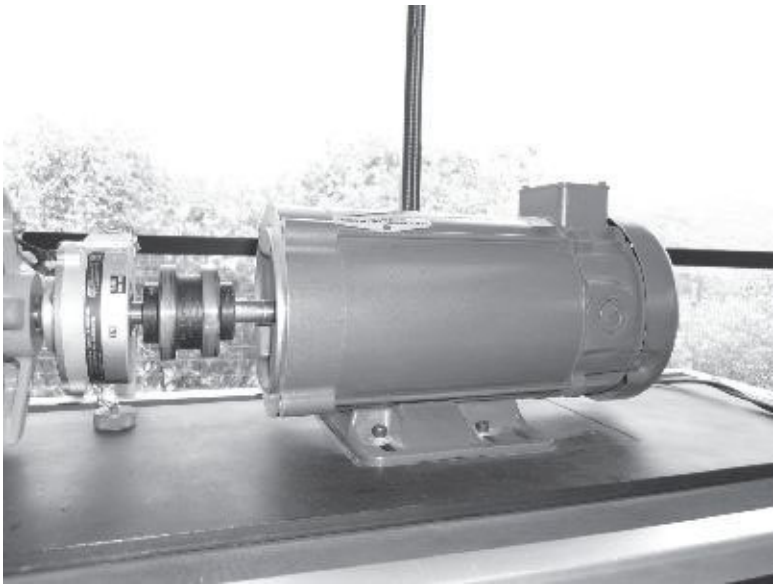

Figure 6.8 Wind turbine emulated by a DC motor.

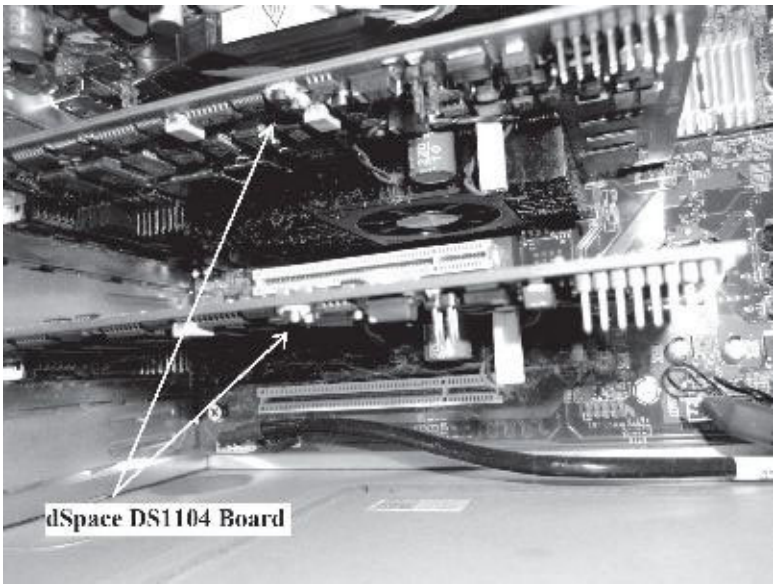

Figure 6.9 DS1104 data acquisition board.

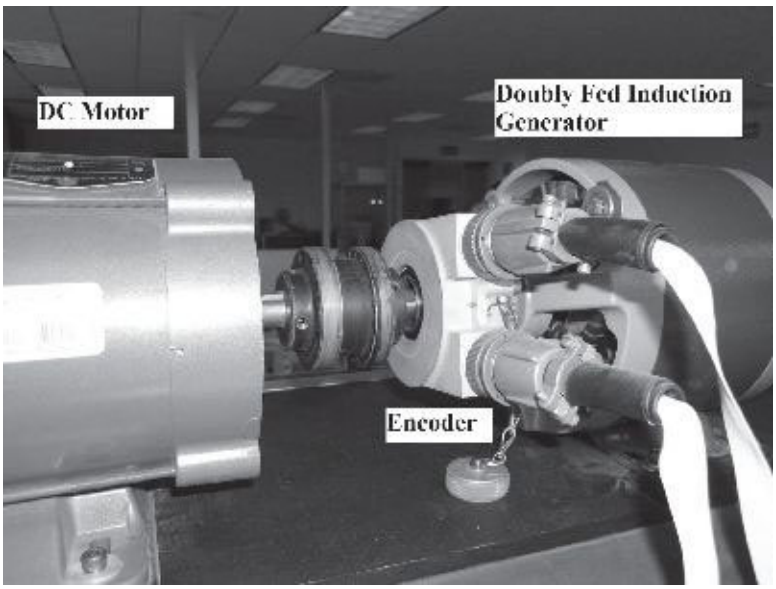

Figure 6.10 Encoder coupled between the DFIG and the DC motor.

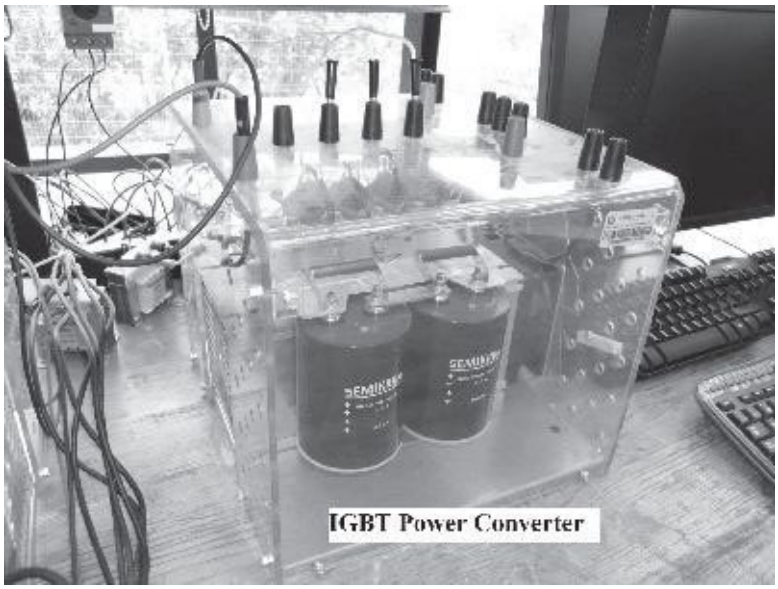

Figure 6.11 PWM driver.

Table 6.2
Control Schemes Implemented in Real Time Successfully

System	Sliding Modes	Inverse Optimal Control	Neural Sliding Modes	Neural Inverse Optimal Control
DFIG	OK	Fail	OK	OK
DC Link	OK	Fail	OK	OK

The real time implementation of the inverse optimal control algorithm (Section 4.2) was not possible due to us knowing the nominal parameters of the prototype, which naturally vary during operation of the generator, and the controller depends strongly on knowledge of these parameters, which cannot be measured in real time. This situation was remedied by reformulating the inverse optimal control based on a neural model, as can be seen in Section 5.3.

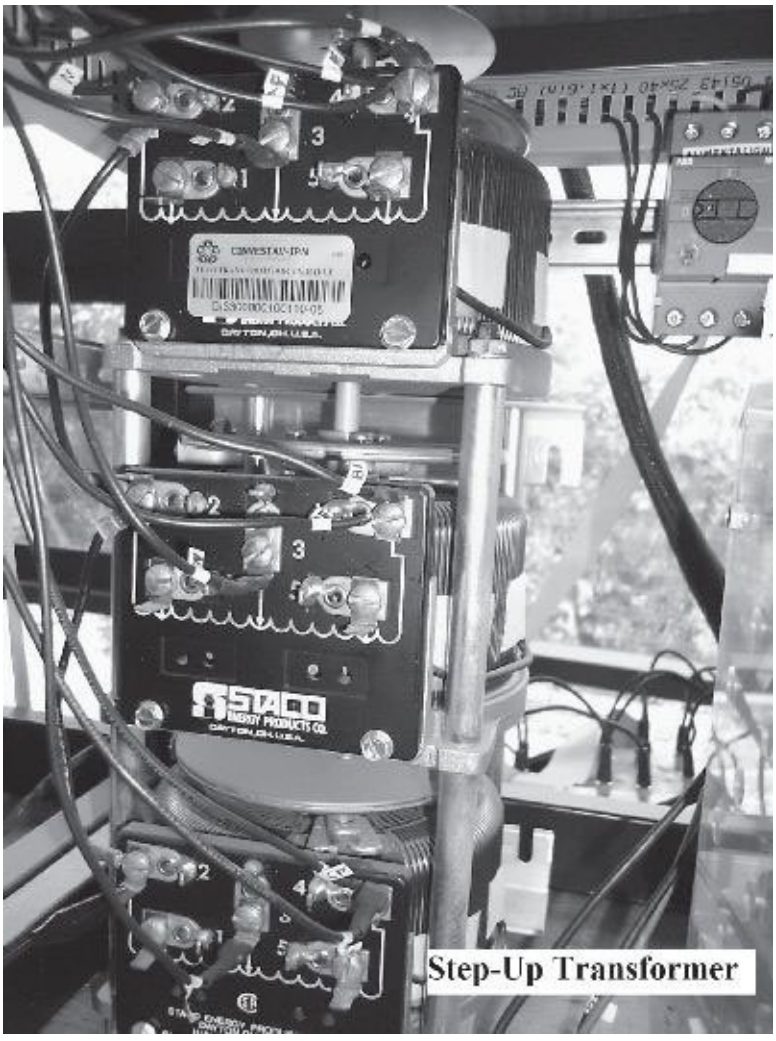

Figure 6.12 Three-phase auto-transformer.

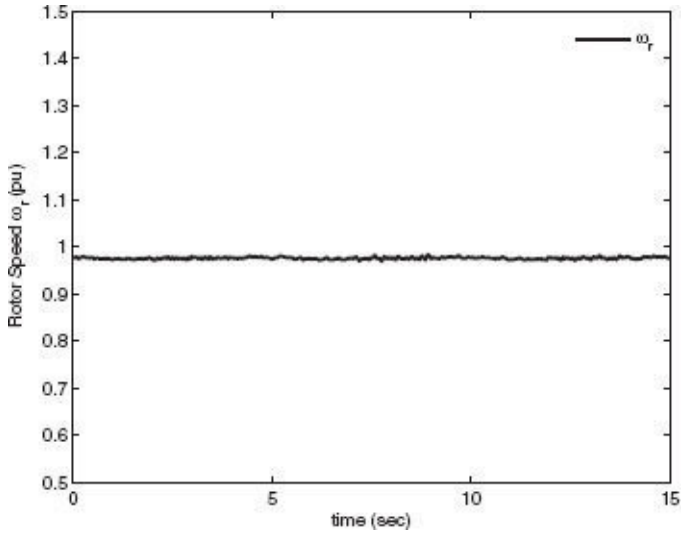

Figure 6.13 Rotor speed (ω_r) imposed by the DC motor.

6.3 SLIDING MODES REAL-TIME RESULTS

In this section, the real time results of the sliding mode controllers designed in Section 4.1 are presented. The real time implementation conditions are:

Capture time: 15 seconds.

Sampling time: $t_s = 0.5\ ms$.

The electric torque reference τ_e^{ref} is a senoidal signal centered at 0.5 *pu* with amplitude of 0.2 *pu* and 0.2 *Hz*.

Power factor reference f_{ps1}^{ref} is constant at 1.0.

The DC voltage reference v_{dc}^{ref} is a constant signal at 0.5567 *pu*.

Power factor reference f_{ps2}^{ref} is constant at 1.0.

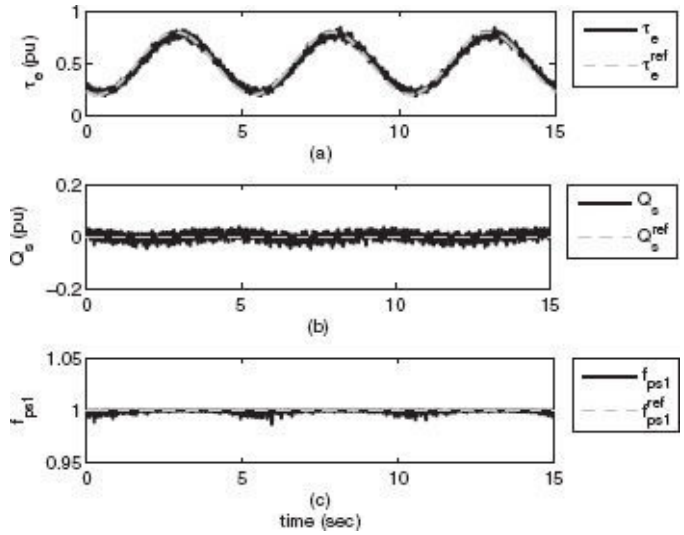

Figure 6.14 DFIG outputs with sliding modes: (a) electric torque (τ_e) tracking, (b) reactive power (Q_s) tracking, and (c) electric power factor (f_{ps1}).

Figure 6.14 presents (a) the electric torque (τ_e) tracking, (b) the reactive power (Q_s)

tracking and (c) electric power factor. In this figure, it can be seen that the tracking for electric torque is achieved despite the reference signal being time variant. In Figure 6.14(b), it can be seen that the reactive power kept almost constant in 0 *pu* slightly affected by the dynamics of the electric torque reference; even so, the power factor (f_{ps1}) is not affected significantly as we can see in Figure 6.14(c). The control signals are bounded and these are shown in Figure 6.15.

The performance of the DC Link controller is shown in the Figure 6.16 and Figure 6.17. The DC voltage (v_{dc}) (Figure 6.16(a)) and reactive power (Q_g) (Figure 6.16(b)) are controlled to the reference. In this figure, we can see that the DC voltage kept in the reference, and the reactive power (Q_g) kept the average value in the reference; it presents small variations due to noise in the measurement of currents and voltages, including also the effect of the switching of the insulated-gate bipolar transistor (IGBT), however, the control objective is achieved. The electric power factor (f_{ps2}) is kept very close to 1. The control signals v_{dg} and v_{qg} are bounded, as shown in Figure 6.17.

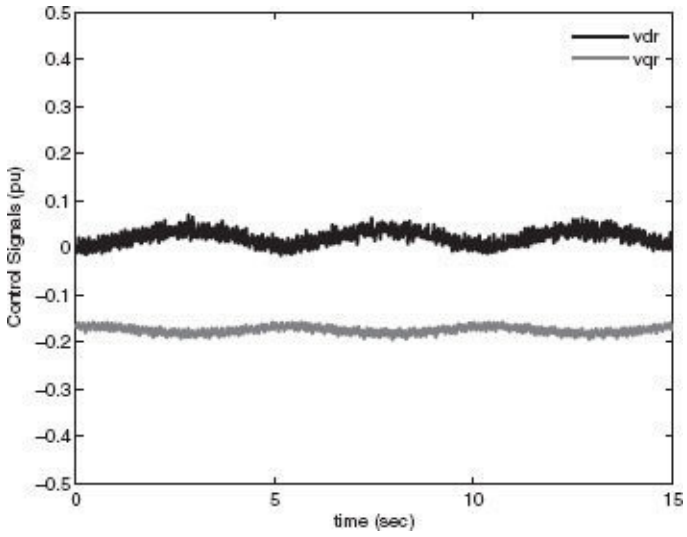

Figure 6.15 DFIG control signals v_{dr} and v_{qr} for sliding modes.

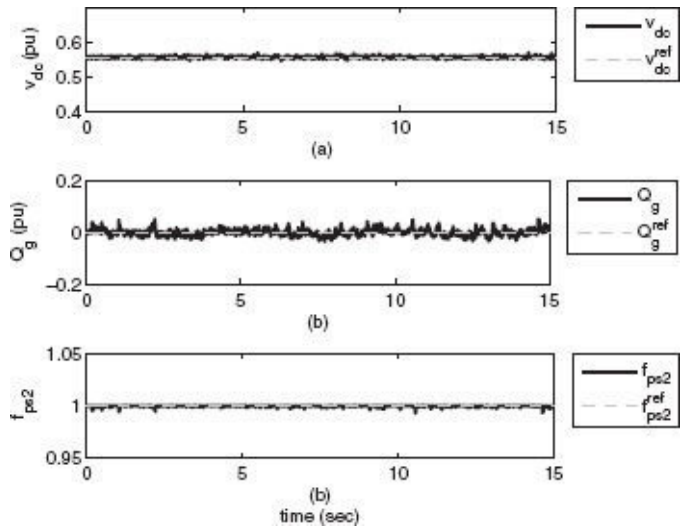

Figure 6.16 DC Link outputs with sliding modes: (a) DC voltage (v_{dc}) tracking, (b) reactive power (Q_g) tracking, and (c) electric power factor (f_{ps2}) in the step-up transformer.

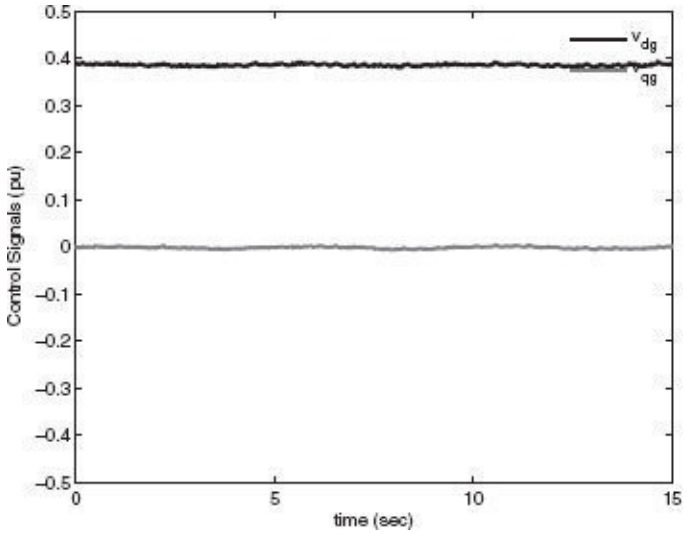

Figure 6.17 DC Link control signals v_{dg} and v_{qg} for sliding modes.

Table 6.3
Statistical Measures of Real Time Implementation Results of the Sliding Modes Controller

Measure	ε_{τ_e}	ε_{Q_s}	$\varepsilon_{f_{ps1}}$	$\varepsilon_{v_{dc}}$	ε_{Q_g}	$\varepsilon_{f_{ps2}}$
MEAN	1.0792e-005	3.3333e-005	-6.8908e-004	6.8583e-006	3.7931e-004	-3.9046e-004
STD	0.0424	0.0146	0.0012	0.0024	0.0128	6.1217e-004
MSE	0.0018	2.1318e-004	1.8404e-006	5.7453e-006	1.6297e-004	5.2719e-007

Quantitative measure of the performance of this real time implementation are shown in Table 6.3, where $\varepsilon_{\tau_e} = \tau_e - \tau_e^{ref}$, $\varepsilon_{Q_s} = Q_s - Q_s^{ref}$, $\varepsilon_{f_{ps1}} = f_{ps1} - f_{ps1}^{ref}$, $\varepsilon_{v_{dc}} = v_{dc}$, and $\varepsilon_{Q_g} = Q_g - Q_g^{ref}$ $- v_{dc}^{ref}$, $\varepsilon_{f_{ps2}} = f_{ps2} - f_{ps2}^{ref}$.

6.4 NEURAL SLIDING MODES REAL-TIME RESULTS

In this section, the real time results of the neural sliding modes controllers designed in Section 5.2 are presented. The real time implementation conditions are:

Capture time: 15 seconds.

Sampling time: $t_s = 0.5\ ms$.

The electric torque reference τ_e^{ref} is a senoidal signal centered at 0.5 *pu* with amplitude of 0.2 *pu* and 0.2 *Hz*.

Power factor reference f_{ps1}^{ref} is constant at 1.0.

The DC voltage reference v_{dc}^{ref} is a constant signal at 0.5567 *pu*.

Power factor reference f_{ps2}^{ref} is constant at 1.0.

Figure 6.18 presents (a) the electric torque (τ_e) tracking, (b) the reactive power (Q_s) tracking, and (c) electric power factor. In this figure, it can be seen that the tracking for electric torque is achieved. In Figure 6.18(b), it can be seen that the reactive power kept almost constant in 0 *pu* was slightly affected by the dynamics of the electric torque reference similar to sliding mode controller; and the power factor (f_{ps1}) variations are more evident compared with the ones of the previous controller, as can be see in Figure 6.18(c). The control signals are bounded and these are shown in Figure 6.19.

The performance of the DC Link neural controller is shown in Figure 6.20. The DC voltage (v_{dc}) is shown in Figure 6.20(a), where we can see that the DC voltage kept close to the reference, but it presents variations more evident. The mean value of the reactive power (Q_g) is in the reference, but it presents variations due to noise in the measurement of currents and voltages, including also the effect of the switching of the IGBT similar to the previous controller, however, the control objective is achieved. It is evident that the electric power factor (f_{ps2}) is kept close to 1 even though there are slight variations in the reactive power. The control signals v_{dg} and v_{qg} are bounded, as shown in Figure 6.21.

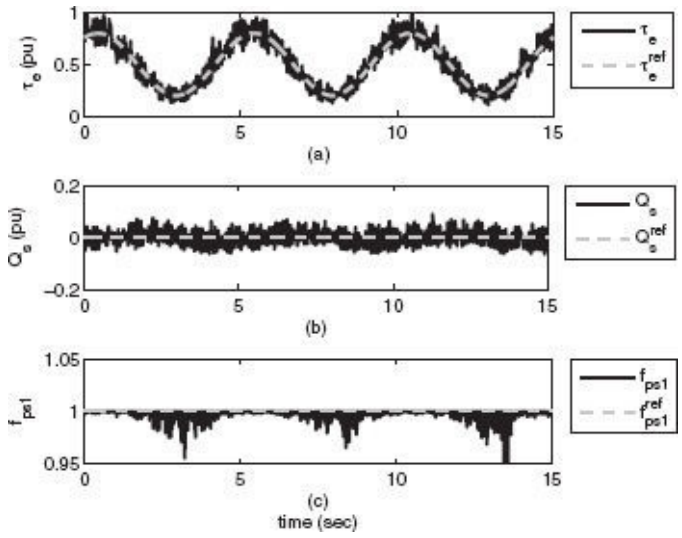

Figure 6.18 DFIG outputs with neural sliding modes: (a) electric torque (τ_e) tracking, (b) reactive power (Q_s) tracking, and (c) electric power factor (f_{ps1}).

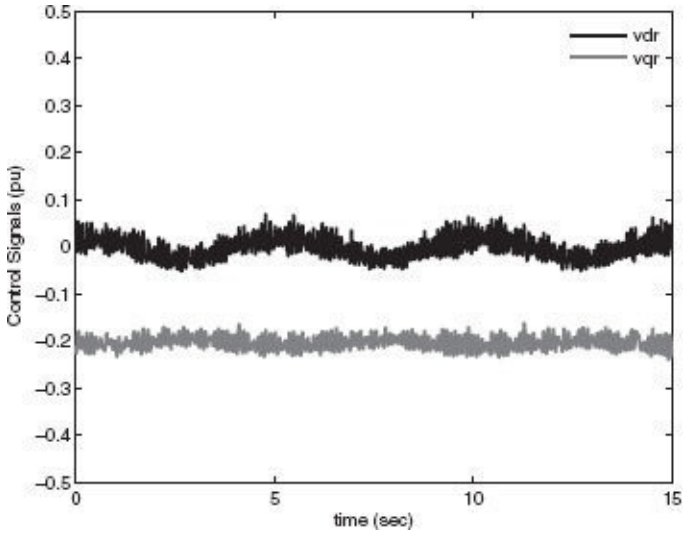

Figure 6.19 DFIG control signals v_{dr} and v_{qr} for neural sliding modes.

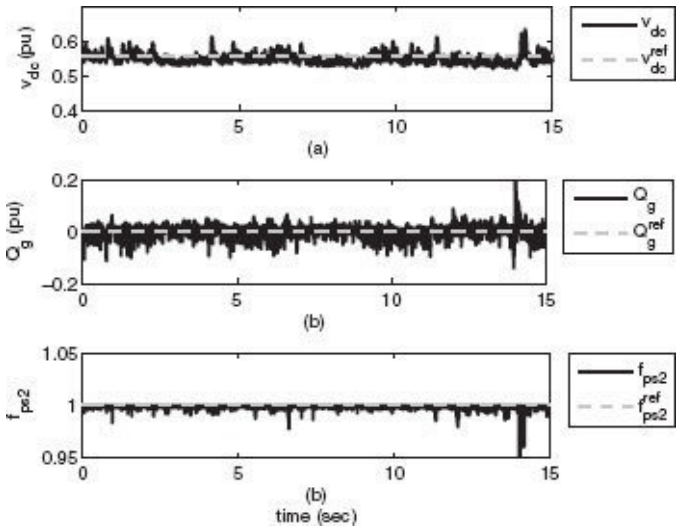

Figure 6.20 DC Link outputs with neural sliding modes: (a) DC voltage (v_{dc}) tracking, (b) reactive power (Q_g) tracking, and (c) electric power factor (f_{ps2}) in the step-up transformer.

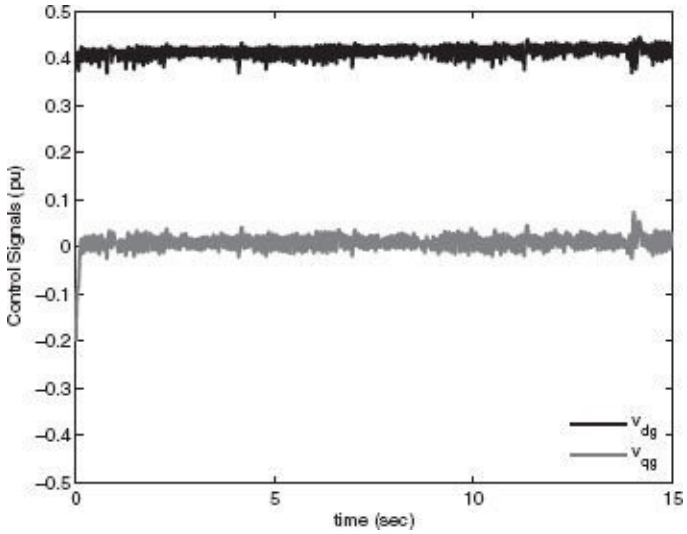

Figure 6.21 DC Link control signals v_{dg} and v_{qg} for neural sliding modes.

Table 6.4

Statistical Measures of Real Time Implementation Results of the Neural Sliding Modes Controller

Measure	ε_{τ_e}	ε_{Q_s}	$\varepsilon_{f_{ps1}}$	$\varepsilon_{v_{dc}}$	ε_{Q_g}	$\varepsilon_{f_{ps2}}$
MEAN	0.0060	-8.5726e-005	-0.0025	-0.0049	0.0063	-0.0024
STD	0.0684	0.0260	0.0045	0.0175	0.0296	0.0118
MSE	0.0047	6.7426e-004	2.6214e-005	3.3153e-004	9.1429e-004	1.4373e-004

Quantitative measures of the performance of this real time implementation are shown in Table 6.4.

6.5 NEURAL INVERSE OPTIMAL CONTROL REAL-TIME RESULTS

In this section, the real time results of the neural inverse optimal controllers designed in Section 5.3 are presented. The real time implementation conditions are:

Capture time: 15 seconds.

Sampling time: $t_s = 0.5\ ms$.

The electric torque reference τ_e^{ref} is a senoidal signal centered at 0.5 *pu* with amplitude of 0.2 *pu* and 0.2 *Hz*.

Power factor reference f_{ps1}^{ref} is constant at 1.0.

The DC voltage reference v_{dc}^{ref} is a constant signal at 0.5567 *pu*.

Power factor reference f_{ps2}^{ref} is constant at 1.0.

Figure 6.22 presents (a) the electric torque (τ_e) tracking, (b) the reactive power (Q_s) tracking, and (c) electric power factor. In this figure, it can be seen that the tracking for electric torque is achieved with a small error. In Figure 6.22(b), it can be seen that the reactive power kept almost constant in 0 *pu* slightly affected by the dynamics of the electric torque reference similar to sliding mode controller; and the power factor (f_{ps1}) variations are small as the sliding modes controller, as can be seen in Figure 6.22(c). The control signals are bounded and these are shown in Figure 6.23.

The performances of the DC Link neural controller and the respective control signals are shown in Figure 6.24 and Figure 6.25, respectively. The DC voltage (v_{dc}) is shown in Figure 6.24(a), where we can see that the DC voltage tracking is very good. The reactive power (Q_g) tracking is achieved successfully, but it presents variations due to noise in the measurement of currents and voltages, including also the effect of the switching of the IGBT similar to previous controllers; however, the control objective is achieved. We can see that the electric power factor (f_{ps2}) is kept close to reference at 1.0. The control signals v_{dg} and v_{qg} are bounded, as shown in Figure 6.25.

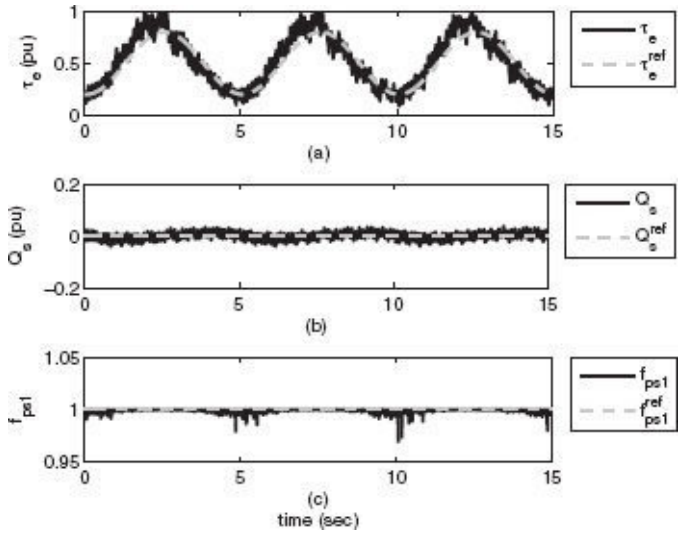

Figure 6.22 DFIG outputs with neural inverse optimal: (a) electric torque (τ_e) tracking, (b) reactive power (Q_s) tracking, and (c) electric power factor (f_{ps1}).

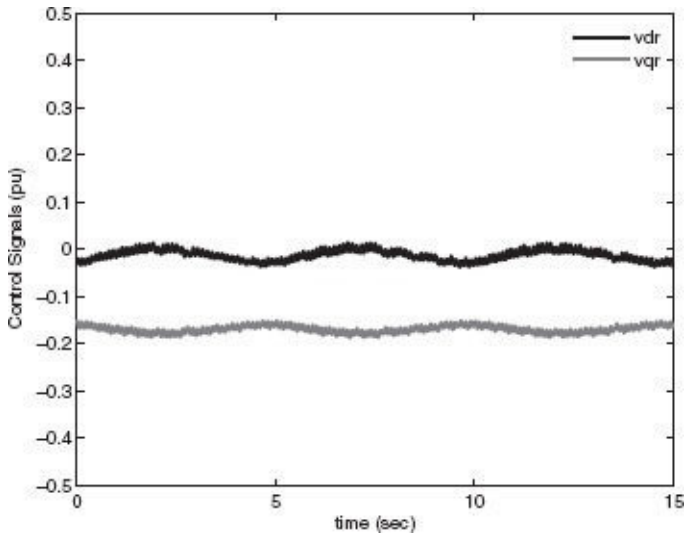

Figure 6.23 DFIG control signals v_{dr} and v_{qr} for neural inverse optimal.

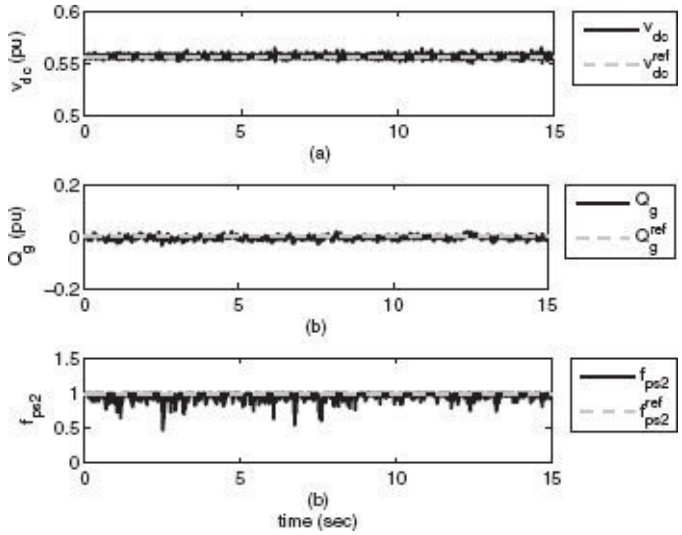

Figure 6.24 DC Link outputs with neural inverse optimal: (a) DC voltage (v_{dc}) tracking, (b) reactive power (Q_g) tracking, and (c) electric power factor (f_{ps2}) in the step-up transformer.

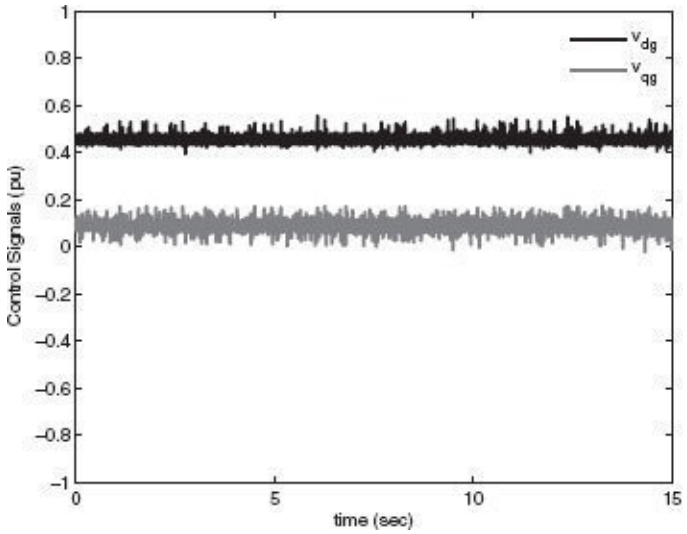

Figure 6.25 DC Link control signals v_{dg} and v_{qg} for neural inverse optimal.

Table 6.5
Statistical Measures of Real Time Implementation Results of the Neural Inverse Optimal Controller

Measure	ε_{τ_e}	ε_{Q_s}	$\varepsilon_{f_{ps1}}$	$\varepsilon_{v_{dc}}$	ε_{Q_g}	$\varepsilon_{f_{ps2}}$
MEAN	0.0160	5.5803e-005	-8.0718e-004	-6.6140e-006	-0.0038	-0.0405
STD	0.0962	0.0139	0.0017	0.0010	0.0070	0.0551
MSE	0.0095	1.9200e-004	3.4869e-006	1.0475e-006	6.2826e-005	0.0047

Table 6.6
Resume Statical Measures

Control Scheme	Measure	ε_{τ_e}	ε_{Q_s}	$\varepsilon_{f_{ps1}}$	$\varepsilon_{v_{dc}}$	ε_{Q_g}	$\varepsilon_{f_{ps2}}$
Sliding	MEAN	**1.08e-5**	3.33e-5	**-6.89e-4**	6.86e-6	**3.79e-4**	-3.90e-4
Modes	STD	**0.0424**	0.0146	**0.0012**	0.0024	**0.0128**	6.12e-4
	MSE	**0.0018**	2.13e-4	**1.84e-6**	5.74e-6	**1.63e-4**	5.27e-7
Neural	MEAN	0.0060	-8.57e-5	-0.0025	-0.0049	0.0063	-0.0024
Sliding	STD	0.0684	0.0260	0.0045	0.0175	0.0296	0.0118
Modes	MSE	0.0047	6.74e-4	2.62e-5	3.31e-4	9.14e-4	1.44e-4
Neural	MEAN	0.0160	**5.58e-5**	-8.07e-4	**-6.61e-6**	-0.0038	-0.0405
Inverse	STD	0.0962	**0.0139**	0.0017	**0.0010**	0.0070	0.0551
Optimal	MSE	0.0095	**1.92e-4**	3.49e-6	**1.05e-6**	6.28e-5	0.0047

Quantitative measures of the performance of this real time implementation are shown in

Table 6.5.

Based on the above real time implementation results, the Table 6.6 resumes the statistical measures to determine the best controller implemented.

As we can see from Table 6.6, the controllers with less tracking error are developed using discrete time sliding mode and the inverse optimal control with neural networks. Although the neural sliding mode controller does not have the lowest tracking error, the convergence time to the reference is smaller than the one of the other controllers. The real time results that make this fact obvious are not presented in this book due to space restrictions.

The main advantage of the neural inverse optimal control is that this algorithm has smoother control signals, which can be seen by comparing the control signals in Figure 6.15, Figure 6.19, and Figure 6.23 for the DFIG, and the respective Figure 6.17, Figure 6.21 and Figure 6.25 for the DC Link.

All results presented in this chapter validate the effectiveness of the algorithms developed in this book.

NOTES

[1] DS1104 R&D Controller Board of dSPACE GmbH.

[2] MATLAB/Simulink® is a trademark of MathWorks.

[3] www.labvolt.com.

[4] www.baldor.com.

[5] DS1104 R&D Controller Board of dSPACE GmbH

[6] MATLAB/Simulink® is a trademark of MathWorks.

Appendix

A Particle Swarm Optimization for Control Algorithms

All the nonlinear control algorithms need to select at least one control parameter, which determines the convergence and performance of the close-loop system. Until now, each nonlinear control algorithm used in this work has its conditions to select the control parameters to ensure the convergence, but a method does not exist to determine the best parameter value according to the performance required in the system.

In this chapter, an optimization algorithm is proposed to select the control parameters in order to minimize a fitness function designed to improve the performance of a controller. The algorithm proposed is the particle swarm optimization (PSO), which realizes a heuristic search within a set of possible solutions. In Section 2.7, the PSO preliminaries are stated. In Section A.1, the PSO algorithm is used to search a Lyapunov P matrix for the inverse optimal control. In Section A.2, the PSO algorithm is applied to select covariance matrices of the extended Kalman filter used to train a neural network. Simulation results are presented to validate the applicability of this optimization algorithm.

A.1 PARTICLE SWARM OPTIMIZATION FOR INVERSE OPTIMAL CONTROL

The PSO algorithm is now reformulated for the inverse optimal control problem. The performance of the inverse optimal control depends on the selection of an appropriate P matrix in equation (2.44). Then, the PSO is used to search that appropriate symmetrical P matrix [71]. The $P[m \times m]$ matrix in general is defined as

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1m} \\ p_{12} & p_{22} & \cdots & p_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1m} & p_{2m} & \cdots & p_{mm} \end{bmatrix} \tag{A.1}$$

The particles population is defined as (2.62), and the P matrix elements compose an individual particle, and is defined as

$$s_j = (p_{11j}, \ldots, p_{1mj}, p_{22j}, \ldots, p_{2mj}, \ldots, p_{m1j}, \ldots, p_{mmj})^T, \tag{A.2}$$

where $j = 1,2,\ldots, N$, with N the number of particles.

The fitness function or objective function $f_{ioc}(s)$[1] is defined as the root mean square of

the output tracking error (RMSE) in the system to be controlled. The PSO algorithm compute is illustrated in Figure A.1. The termination criterion can be choice as the number of iterations is reached, the search time limit is exceeded, or the objective function $f_{ioc}(s) \leq \varepsilon$, $\varepsilon > 0$.

A.1.1 DFIG AND DC LINK APPLICATION

The performance of the inverse optimal control is improved using the PSO to select all values of the matrix P_1 in the doubly fed induction generator (DFIG) controller (4.63) and the matrix P_2 in the DC Link controller (4.81).

The matrix P_1 is symmetrical and defined as

$$P_1 = \begin{bmatrix} p_{11(1)} & p_{12(1)} & p_{13(1)} & p_{14(1)} \\ p_{12(1)} & p_{22(1)} & p_{23(1)} & p_{24(1)} \\ p_{13(1)} & p_{23(1)} & p_{33(1)} & p_{34(1)} \\ p_{14(1)} & p_{24(1)} & p_{34(1)} & p_{44(1)} \end{bmatrix}, \tag{A.3}$$

and the PSO is used to calculated $p_{11(1)}$, $p_{12(1)}$, $p_{13(1)}$, $p_{14(1)}$, $p_{22(1)}$, $p_{23(1)}$, $p_{24(1)}$, $p_{33(1)}$, $p_{34(1)}$, and $p_{44(1)}$ such that the tracking error $i_e(k)$ is reduced.

For the rotor side converter (RSC) controller, the fitness function $F_{fit}^{(RSC)}$ to minimize using PSO is defined as

$$F_{fit}^{(RSC)} = \frac{\sum_{j=1}^{M_1} \left(\left(\sum_{k=1}^{N_1} i_e(j,k)^2 \right) / N_1 \right)}{M_1}, \tag{A.4}$$

where M_1 is the number of outputs to control by the RSC and N_1 is the samples number in the simulation.

The matrix P_2 is symmetrical and defined as

$$P_2 = \begin{bmatrix} p_{11(2)} & p_{12(2)} & p_{13(2)} \\ p_{12(2)} & p_{22(2)} & p_{23(2)} \\ p_{13(2)} & p_{23(2)} & p_{33(2)} \end{bmatrix}, \tag{A.5}$$

and the PSO is used to calculate the $p_{11(2)}$, $p_{12(2)}$, $p_{13(2)}$, $p_{22(2)}$, $p_{23(2)}$, and $p_{33(2)}$ such that the tracking error ε_{x_g} is reduced.

For the grid side converter (GSC) controller, the fitness function $F_{fit}^{(GSC)}$ to minimize

using PSO is defined as

$$F_{fit}^{(GSC)} = \frac{\sum_{j=1}^{M_2}\left(\left(\sum_{k=1}^{N_2}\varepsilon_{x_g}(j,k)^2\right)/N_2\right)}{M_2}, \tag{A.6}$$

where M_2 is the number of outputs to control by the GSC and N_2 is the samples number in the simulation.

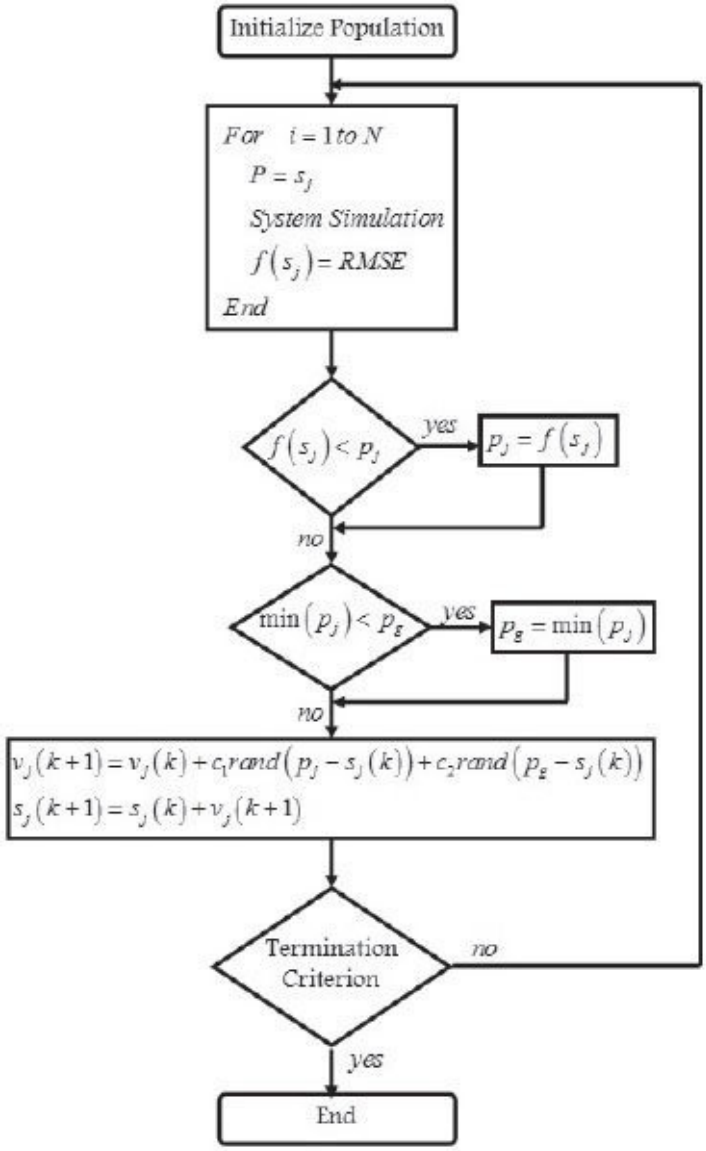

Figure A.1 PSO flow diagram.

A.1.1.1 Simulation Results

To evaluate the performance of the developed controllers, simulations are performed. The DFIG is a three-phase generator, four poles, with a stator-referred rotor. The generator parameters appear in Table A.1.

Table A.1

Doubly Fed Induction Generator and DC Link Parameters

Symbol	Parameter	Value
X_m	Magnetizing Reactance	2.3175 *pu*
X_s	Stator Reactance	2.4308 *pu*
X_r	Rotor Reactance	2.4308 *pu*
r_s	Stator Windings Resistance	0.1609 *pu*
r_r	Rotor Windings Resistance	0.0502 *pu*
H	Angular Moment of Inertia	0.23 *sec*
ω_b	Base Angular Frequency	376.99112 *rad/sec*
P_b	Base Power	185.4 *VA*
V_b	Base Voltage	179.63 *V*
X_l	Three Phase Lines Reactance	0.0045 *pu*
r_g	Three Phase Lines Resistance	0.0014 *pu*
C	DC Link Capacitance	0.1854 *pu*

For both controllers, 8 particles and 100 iterations are selected. Table A.2 and Table A.3 show the initial parameters $P_{1(0)}$ for the inverse optimal control (IOC) and the $P_{1(PSO)}$ determined for the inverse optimal control (PSOIOC) applying the PSO algorithm.

For the RSC controller the initial fitness function is $\boldsymbol{F_{fit(0)}^{(RSC)} = 0.0144}$ and the fitness function using PSO is $\boldsymbol{F_{fit(PSO)}^{(RSC)} = 0.0113}$.

Table A.2
RSC Inverse Optimal Control Parameters (Part I)

Control	$p_{11(1)}$	$p_{12(1)}$	$p_{13(1)}$	$p_{14(1)}$	$p_{22(1)}$
IOC	1	0.5	0	0	7
PSOIOC	54.65	−10.45	21.81	−33.02	477.57

The electric torque and the stator reactive power are shown in Figure A.2 and Figure A.3, respectively. The establish time is smaller using the parameters found with the PSO. The electric torque peak for the PSOIOC is less than the one obtained by IOC. The generator currents are shown in Figure A.4. The transient process for the currents is improved using the PSO. In Figure A.5, the control signals for the RSC are shown. The control signals for the IOC are smaller than the ones for the PSOIOC, but are bounded. This is because the control signals are not included in the fitness function $\boldsymbol{F_{fit}^{(RSC)}}$.

Table A.3
RSC Inverse Optimal Control Parameters (Part II)

Control	$p_{23(1)}$	$p_{24(1)}$	$p_{33(1)}$	$p_{34(1)}$	$p_{44(1)}$
IOC	0.5	0	1	0.5	7
PSOIOC	1.78	−56.75	68.00	−3.91	839.43

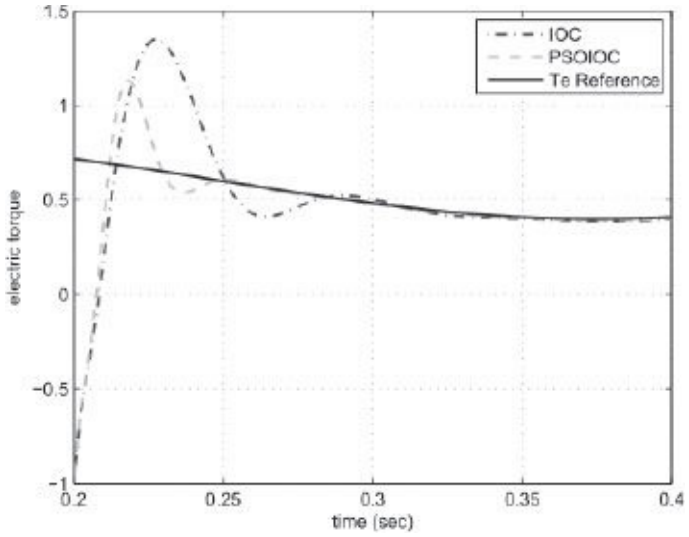

Figure A.2 Electric torque.

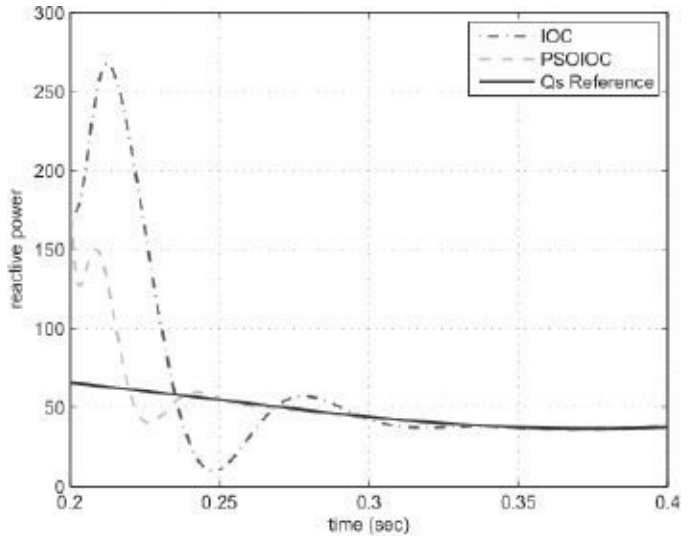

Figure A.3 Reactive power.

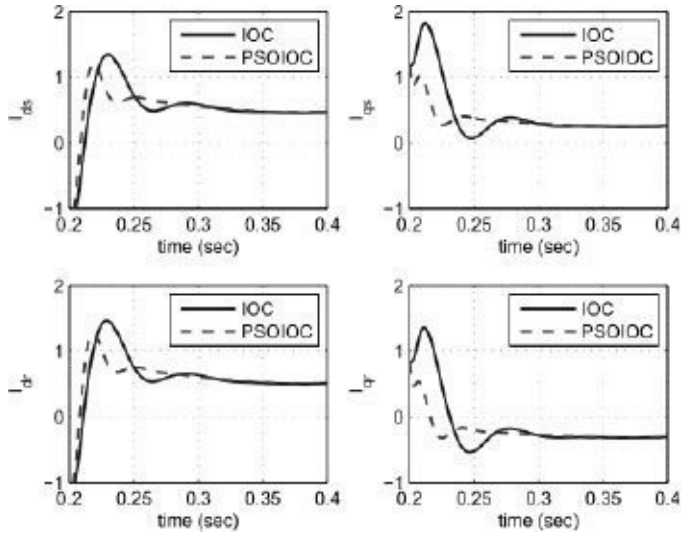

Figure A.4 Stator and rotor currents.

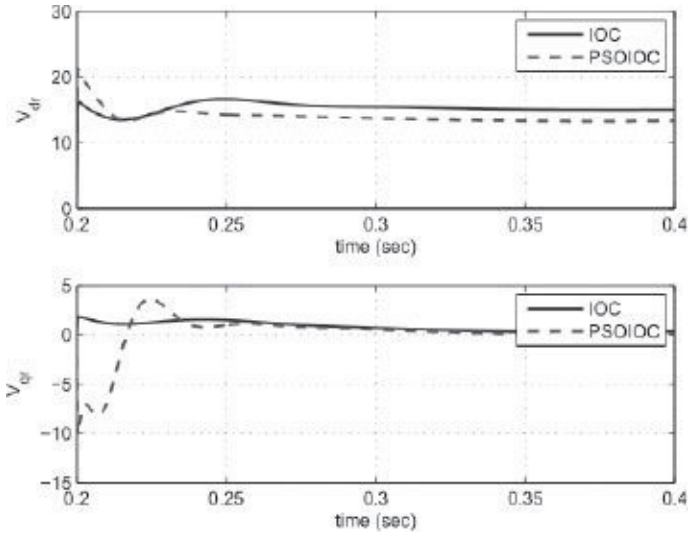

Figure A.5 Control signals v_{dr} and v_{qr}.

Similarly, Table A.4, shows the initial $P_{2(0)}$ and the $P_{2(PSO)}$ found applying the PSO.

Table A.4
GSC Inverse Optimal Control Parameters

Control	$p_{11(2)}$	$p_{12(2)}$	$p_{13(2)}$	$p_{22(2)}$	$p_{23(2)}$	$p_{33(2)}$
IOC	0.5	0.25	0	0.5	0.25	0.5
IOCPSO	0.40	0.05	0.25	1.13	0.49	0.63

The DC Link tracking using IOC and PSOIOC are displayed in Figure A.6. The transient process with PSOIOC is faster than IOC. The current tracking i_{dg} and i_{qg} are displayed in Figure A.7 and Figure A.8. The control signals for the GSC are displayed in Figure A.9 and Figure A.10. In these figures it is easy to see that the control signals using PSOIOC are smaller than the ones with IOC. For this application the performance improvement is very important, because the transient process is faster with the PSOIOC.

A.2 PARTICLE SWARM OPTIMIZATION FOR NEURAL NETWORKS

Until now, the neural networks trained using Kalman filter have had a very good performance in different applications. The convergence criterion of the Kalman filter in neural networks was investigated in [7]. However, the selection procedure of the covariance matrices is not established formally. This selection depends on experience of the designer.

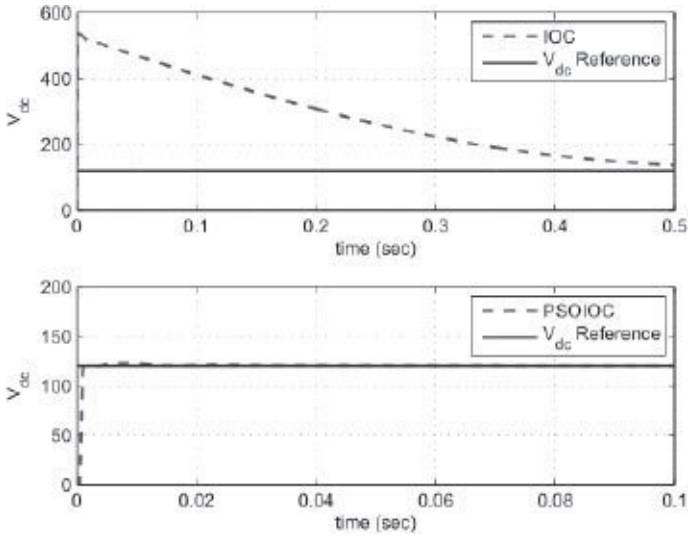

Figure A.6 DC Link voltage V_{dc}.

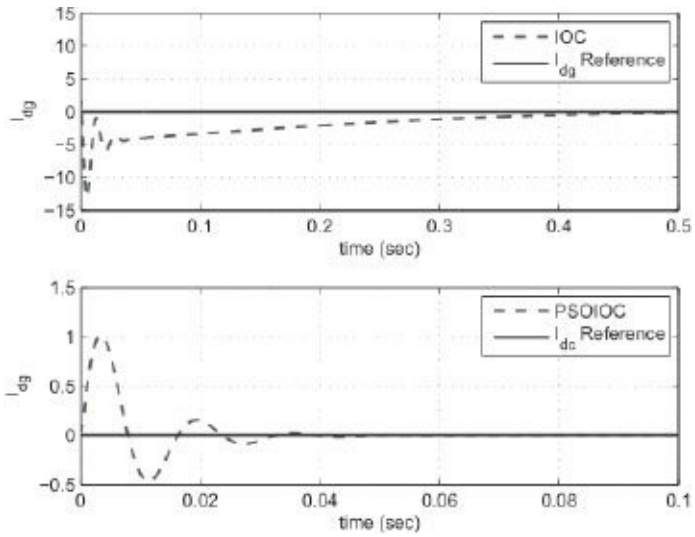

Figure A.7 Current i_{dg}.

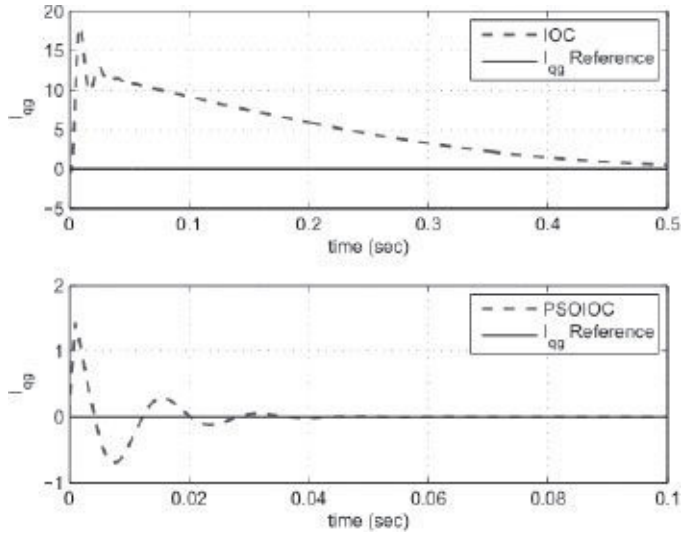

Figure A.8 Current i_{qg}.

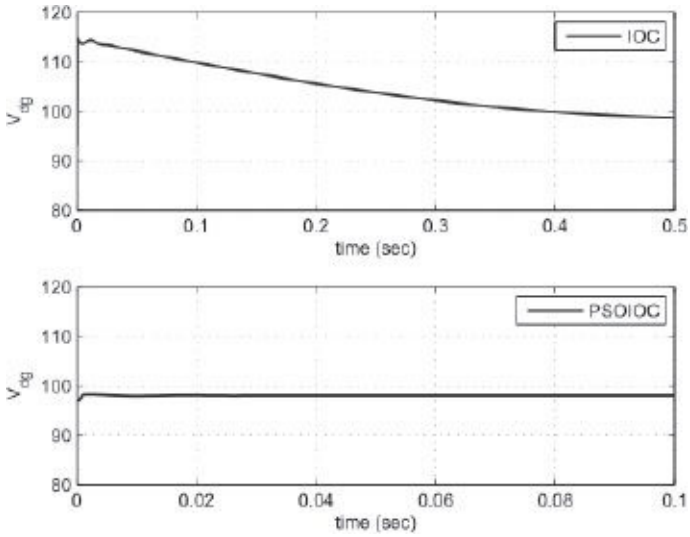

Figure A.9 Control signal v_{dg}.

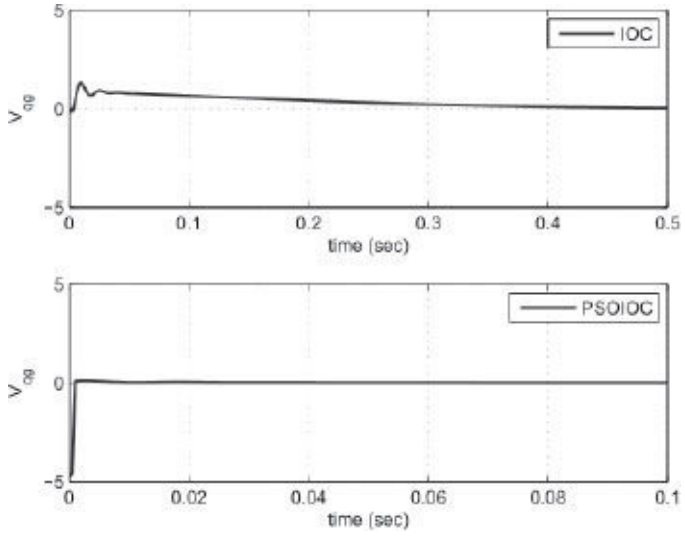

Figure A.10 Control signal v_{qg}.

In this section, a PSO algorithm is proposed as an option to find the Kalman filter covariance matrices P, R and Q in (2.59).

Usually the matrices P_i, Q_i, and R_i are initialized as diagonal matrices, with entries $P_i(0)$, $Q_i(0)$, and $R_i(0)$, respectively. It is important to note that $H_i(k)$, $K_i(k)$, and $P_i(k)$ for the extended Kalman filter (EKF) are bounded [76]. Then, the PSO is used to search those appropriate $P_i(0)$, $Q_i(0)$, and $R_i(0)$ values.

The particles population is defined as (2.62), and the $P_i(0)$, $Q_i(0)$, and $R_i(0)$ values compose an individual particle, defined as

$$s_j = \left(P_{ij}\left(0\right), Q_{ij}\left(0\right), R_{ij}\left(0\right)\right)^T, \tag{A.7}$$

where $j = 1,2,\ldots, N$, with N the number of particles.

The fitness function or objective function $F_{nn}(s)$[2] is defined as the RMSE in the system to be controlled, as follows:

$$F_{nn}\left(s\right) = \frac{\sum_{j=1}^{M_{nn}} \left(\left(\sum_{k=1}^{N_{nn}} \left(y\left(j,k\right) - y^{ref}\left(j,k\right)\right)^2 \right) / N_{nn} \right)}{M_{nn}}, \tag{A.8}$$

where M_{nn} is the number of outputs to control by the recurrent high-order neural network (RHONN) and N_{nn} is the sample number in the simulation.

The PSO algorithm computed is illustrated in Figure A.1. The termination criterion can be choice as the number of iterations is reached, the search time limit is exceeded, or the objective function $F_{nn}(s) \leq \varepsilon$, $\varepsilon > 0$.

NOTES

[1] The subscript *ioc* refers to inverse optimal control.

[2] The subscript *nn* refers to neural network.

B DFIG Modeling

B.1 DFIG MATHEMATICAL MODEL

The induction machine is used in a wide variety of applications as a means of converting electric power to mechanical work or reverse. The voltage equations for the elementary induction machine shown in Figure B.1 and Figure B.2 are [38]

$$\begin{aligned} v_{as} &= r_s i_{as} + \frac{d\lambda_{as}}{dt}, \\ v_{bs} &= r_s i_{bs} + \frac{d\lambda_{bs}}{dt}, \\ v_{cs} &= r_s i_{cs} + \frac{d\lambda_{cs}}{dt}, \\ v_{ar} &= r_r i_{ar} + \frac{d\lambda_{ar}}{dt}, \\ v_{br} &= r_r i_{br} + \frac{d\lambda_{br}}{dt}, \\ v_{cr} &= r_r i_{cr} + \frac{d\lambda_{cr}}{dt}, \end{aligned} \tag{B.1}$$

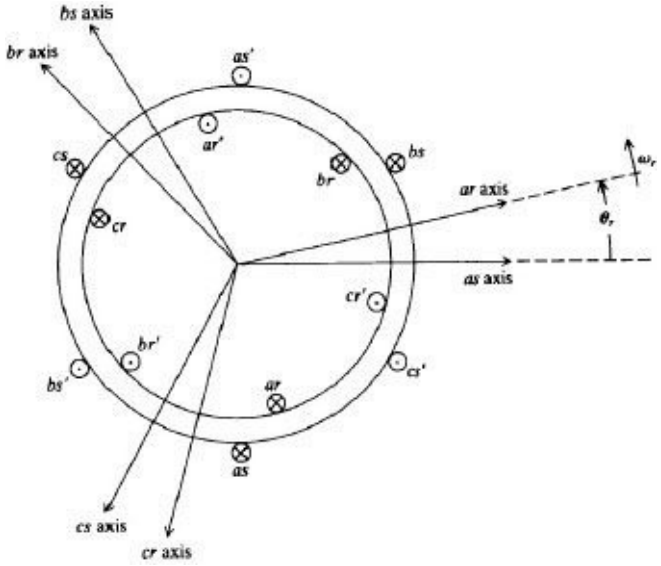

Figure B.1 Two-pole, 3-phase, elementary induction machine.

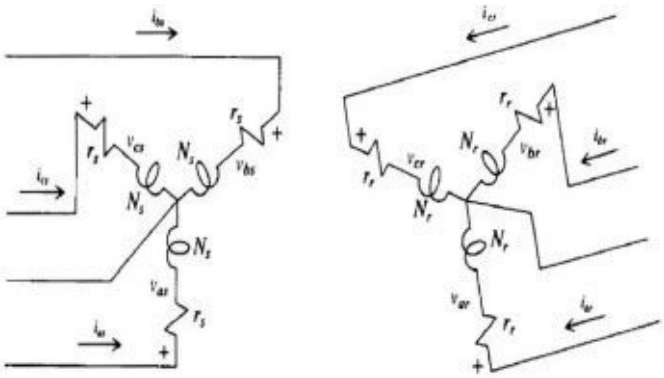

Figure B.2 Direction currents induction motor convention.

where v_{as}, v_{bs}, v_{cs} are the stator voltages in the *a, b, c* axis, respectively, i_{as}, i_{bs}, i_{cs} are the stator currents, v_{ar}, v_{br}, v_{cr} are the rotor voltages, i_{ar}, i_{br}, i_{cr} are the rotor currents, r_s is the resistance of the stator winding, r_r is the resistance of the rotor winding. The flux linkages are expressed as

$$\begin{aligned}
\lambda_{as} &= L_{asas}i_{as} + L_{asbs}i_{bs} + L_{ascs}i_{cs} + L_{asar}i_{ar} + L_{asbr}i_{br} + L_{ascr}i_{cr},\\
\lambda_{bs} &= L_{bsas}i_{as} + L_{bsbs}i_{bs} + L_{bscs}i_{cs} + L_{bsar}i_{ar} + L_{bsbr}i_{br} + L_{bscr}i_{cr},\\
\lambda_{cs} &= L_{csas}i_{as} + L_{csbs}i_{bs} + L_{cscs}i_{cs} + L_{csar}i_{ar} + L_{csbr}i_{br} + L_{cscr}i_{cr},\\
\lambda_{ar} &= L_{aras}i_{as} + L_{arbs}i_{bs} + L_{arcs}i_{cs} + L_{arar}i_{ar} + L_{arbr}i_{br} + L_{arcr}i_{cr},\\
\lambda_{br} &= L_{bras}i_{as} + L_{brbs}i_{bs} + L_{brcs}i_{cs} + L_{brar}i_{ar} + L_{brbr}i_{br} + L_{brcr}i_{cr},\\
\lambda_{cr} &= L_{cras}i_{as} + L_{crbs}i_{bs} + L_{crcs}i_{cs} + L_{crar}i_{ar} + L_{crbr}i_{br} + L_{crcr}i_{cr},
\end{aligned} \tag{B.2}$$

The winding inductances of the induction machine may be expressed from the inductance relationships given for the salient-pole synchronous machine. In the case of the induction machine the air gap is uniform. All stator self-inductances are equal; that is, $L_{asas} = L_{bsbs} = L_{cscs}$ with

$$L_{asas} = L_{ls} + L_{ms}, \tag{B.3}$$

where L_{ms} is the stator magnetizing inductance. Likewise all stator-to-stator mutual inductances are the same.

$$L_{asbs} = L_{bscs} = L_{csas} = -\frac{L_{ms}}{2}. \tag{B.4}$$

In Figure B.1, the *ar* axis is displaced with *as* axes at an angle θ_r. Similarly, the axes *br, cr* are displaced with the axes *bs, cs*, respectively, at the same angle θ_r. So the stator-to-rotor mutual inductances are defined as

$$L_{asar} = L_{bsbr} = L_{cscr} = \frac{N_r}{N_s}L_{ms}\cos(\theta_r). \tag{B.5}$$

The angle between the axis *as* and *br* is $\theta_r + \frac{2\pi}{3}$; then

$$L_{asbr} = L_{bscr} = L_{csar} = \frac{N_r}{N_s}L_{ms}\cos\left(\theta_r + \frac{2\pi}{3}\right). \tag{B.6}$$

The *as* axis is displaced with *cr* at an angle $\theta_r - \frac{2\pi}{3}$; then

$$L_{ascr} = L_{bsar} = L_{csbr} = \frac{N_r}{N_s} L_{ms} cos\left(\theta_r - \frac{2\pi}{3}\right). \tag{B.7}$$

All rotor-to-rotor mutual inductances are the same and are defined as

$$L_{arar} = L_{brbr} = L_{crbr} = L_{lr} + \left(\frac{N_r}{N_s}\right)^2 L_{ms}, \tag{B.8}$$

where L_{lr} is the rotor leakage inductance. Finally, the mutual inductances between *ar* and *br, br* and *cr*, and *cr* and *ar* are defined in terms of the stator mutual inductances as

$$L_{arbr} = L_{brcr} = L_{crar} = -\left(\frac{N_r}{N_s}\right)^2 \frac{L_{ms}}{2}. \tag{B.9}$$

In order to simplify the handling of the equations (B.1) and (B.2), it can be rewritten in a matrix form as

$$\begin{aligned} v_{abcs} &= R_s i_{abcs} + \frac{d\lambda_{abcs}}{dt}, \\ v_{abcr} &= R_r i_{abcr} + \frac{d\lambda_{abcr}}{dt}, \\ \lambda_{abcs} &= L_{ss} i_{abcs} + L_{sr} i_{abcr}, \\ \lambda_{abcr} &= L_{st}^T i_{abcs} + L_{rr} i_{abcr}, \end{aligned} \tag{B.10}$$

where

$$f_{abcx} = \begin{bmatrix} f_{ax} \\ f_{bx} \\ f_{cx} \end{bmatrix}. \tag{B.11}$$

The symbol f is used to represent the voltages, currents, and coupling fluxes; the x subscript is used to refer to the stator or rotor. In addition

$$R_s = \begin{bmatrix} r_s & 0 & 0 \\ 0 & r_s & 0 \\ 0 & 0 & r_s \end{bmatrix}, R_r = \begin{bmatrix} r_r & 0 & 0 \\ 0 & r_r & 0 \\ 0 & 0 & r_r \end{bmatrix},$$

$$L_{ss} = \begin{bmatrix} L_{ls} + L_{ms} & -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & L_{ls} + L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} & L_{ls} + L_{ms} \end{bmatrix},$$

$$L_{rr} = \begin{bmatrix} L_{lr} + L_{mr} & -\frac{1}{2}L_{mr} & -\frac{1}{2}L_{mr} \\ -\frac{1}{2}L_{mr} & L_{lr} + L_{mr} & -\frac{1}{2}L_{mr} \\ -\frac{1}{2}L_{mr} & -\frac{1}{2}L_{mr} & L_{lr} + L_{mr} \end{bmatrix},$$

$$L_{sr} = \begin{bmatrix} L_{sr}\cos(\theta_r) & L_{sr}\cos\left(\theta_r + \frac{2\pi}{3}\right) & L_{sr}\cos\left(\theta_r - \frac{2\pi}{3}\right) \\ L_{sr}\cos\left(\theta_r - \frac{2\pi}{3}\right) & L_{sr}\cos(\theta_r) & L_{sr}\cos\left(\theta_r + \frac{2\pi}{3}\right) \\ L_{sr}\cos\left(\theta_r + \frac{2\pi}{3}\right) & L_{sr}\cos\left(\theta_r - \frac{2\pi}{3}\right) & L_{sr}\cos(\theta_r) \end{bmatrix},$$

where

$$L_{mr} = \left(\frac{N_r{}^2}{N_s}\right) L_m s,$$

$$L_{sr} = \frac{N_r}{N_s} L_{ms}.$$

When the voltage equations are expressed as (B.10), it is convenient to refer all rotor variables to stator means the tip ratio N_s/N_r; then

$$i'_{abcr} = \frac{N_r}{N_s} i_{abcr}, \tag{B.12}$$

$$\lambda'_{abcr} = \frac{N_s}{N_r} \lambda_{abcr}, \tag{B.13}$$

$$v'_{abcr} = \frac{N_s}{N_r} v_{abcr}, \tag{B.14}$$

$$r'_r = \left(\frac{N_s}{N_r}\right)^2 r_r, \tag{B.15}$$

$$L'_{lr} = \left(\frac{N_s}{N_r}\right)^2 L_{lr}. \tag{B.16}$$

Using the equations (B.12) to (B.16), then, the equation system (B.10) can be rewritten as

$$\begin{aligned} v_{abcs} &= R_s i_{abcs} + \frac{d\lambda_{abcs}}{dt} \\ v'_{abcr} &= R'_r i'_{abcr} + \frac{d\lambda'_{abcr}}{dt}, \\ \lambda_{abcs} &= L_{ss} i_{abcs} + L'_{sr} i'_{abcr}, \\ \lambda'_{abcr} &= {L'_{sr}}^T i_{abcs} + L'_{rr} i'_{abcr}, \end{aligned} \tag{B.17}$$

where

$$R'_r = \begin{bmatrix} r'_r & 0 & 0 \\ 0 & r'_r & 0 \\ 0 & 0 & r_r \end{bmatrix},$$

$$L'_{rr} = \begin{bmatrix} L'_{lr} + L_{ms} & -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & L_{lr} + L_{ms} & -\frac{1}{2}L_{ms} \\ -\frac{1}{2}L_{ms} & -\frac{1}{2}L_{ms} & L'_{lr} + L_{ms} \end{bmatrix},$$

$$L'_{sr} = \begin{bmatrix} L_{ms}\cos(\theta_r) & L_{ms}\cos\left(\theta_r + \frac{2\pi}{3}\right) & L_{ms}\cos\left(\theta_r - \frac{2\pi}{3}\right) \\ L_{ms}\cos\left(\theta_r - \frac{2\pi}{3}\right) & L_{ms}\cos(\theta_r) & L_{ms}\cos\left(\theta_r + \frac{2\pi}{3}\right) \\ L_{ms}\cos\left(\theta_r + \frac{2\pi}{3}\right) & L_{ms}\cos\left(\theta_r - \frac{2\pi}{3}\right) & L_{ms}\cos(\theta_r) \end{bmatrix},$$

The doubly fed induction generator (DFIG) mathematical model (B.17) considers a direction currents convention as the motor shown in Figure B.2. In this book, the direction currents convention selected is shown in Figure B.3. Then the DFIG mathematical model (B.17) is rewritten as

$$\begin{aligned} v_{abcs} &= -R_s i_{abcs} + \frac{d\lambda_{abcs}}{dt}, \\ v'_{abcr} &= R'_r i'_{abcr} + \frac{d\lambda'_{abcr}}{dt}, \\ \lambda_{abcs} &= -L_{ss} i_{abcs} + L'_{sr} i'_{abcr}, \\ \lambda'_{abcr} &= -{L'_{sr}}^T i_{abcs} + L'_{rr} i'_{abcr}. \end{aligned} \tag{B.18}$$

B.1.1 CIRCUIT VARIABLES TRANSFORMATION REFERRED TO A REFERENCE FRAME FIXED IN THE ROTOR

The voltage equations that describe the performance of induction machines are functions of the rotor speed, whereupon the coefficients of the differential equations that describe the behavior of these machines are time-varying except when the rotor is stalled. A change of variables is often used to reduce the complexity of these differential equations. A general transformation refers machine variables to a frame of reference that rotates at an arbitrary angular velocity. All known real transformations are obtained from this transformation by simply assigning the speed of the rotation of the reference frame [38].

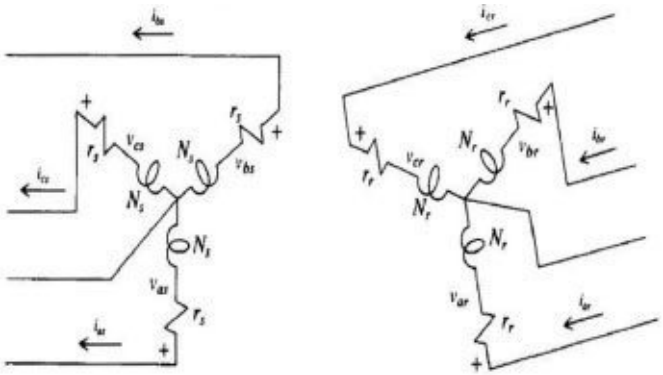

Figure B.3 Direction currents DFIG convention.

The time-varying inductances of a synchronous machine are eliminated only if the reference frame is fixed in the rotor [38], which is called *qd* transformation.

A change of variables that formulates a transformation of the three phase variables of the stator circuits to the arbitrary reference frame is expressed by

$$K_s = \frac{2}{3}\begin{bmatrix} \cos\theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ -\sin\theta & -\sin\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \tag{B.19}$$

where $\omega = \frac{d\theta}{dt}$ is the angular speed of the reference frame. However, in the analysis of induction machines it is also desirable to transform the variables associated with the symmetrical rotor windings to the arbitrary reference frame. A change of variables that formulates a transformation of the three phase variables of the rotor circuits to the arbitrary reference frame is:

$$K_r = \frac{2}{3}\begin{bmatrix} \cos\beta & \cos\left(\beta - \frac{2\pi}{3}\right) & \cos\left(\beta + \frac{2\pi}{3}\right) \\ -\sin\beta & -\sin\left(\beta - \frac{2\pi}{3}\right) & \cos\left(\beta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \tag{B.20}$$

with

$$\beta = \theta - \theta_r.$$

So, the stator and rotor variables transformed using the transformation (B.19) and (B.20) are defined, respectively, as

$$f_{dq0s} = K_s f_{abcs}, \tag{B.21}$$

$$f_{dq0r} = K_r f_{abcr}, \tag{B.22}$$

where the symbol f is used to refer to each DFIG variable.

Applying the transformation (B.19) and (B.20) to the DFIG equations (B.18), the following equations are obtained:

$$\begin{aligned}
K_s^{-1} v_{dq0s} &= -R_s K_s^{-1} i_{dq0s} + \tfrac{d}{dt}\left[K_s^{-1}\lambda_{dq0s}\right], \\
K_r^{-1} v'_{dq0r} &= R'_r K_r^{-1} i'_{dq0r} + \tfrac{d}{dt}\left[K_r^{-1}\lambda'_{dq0r}\right], \\
K_s^{-1}\lambda_{dq0s} &= -L_{ss} K_s^{-1} i_{dq0s} + L'_{sr} K_r^{-1} i'_{dq0r}, \\
K_r^{-1}\lambda'_{dq0r} &= -L'^{T}_{sr} K_s^{-1} i_{dq0s} + L'_{rr} K_r^{-1} i'_{dq0r},
\end{aligned} \tag{B.23}$$

where

$$K_s^{-1} = \begin{bmatrix} \cos\theta & -\sin\theta & 1 \\ \cos\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\theta + \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) & 1 \end{bmatrix}, \tag{B.24}$$

$$K_r^{-1} = \begin{bmatrix} \cos\beta & -\sin\beta & 1 \\ \cos\left(\beta - \frac{2\pi}{3}\right) & -\sin\left(\beta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\beta + \frac{2\pi}{3}\right) & -\sin\left(\beta + \frac{2\pi}{3}\right) & 1 \end{bmatrix}, \tag{B.25}$$

with $\beta = \theta - \theta_r$. Developing (B.23) and reordering terms,

$$\begin{aligned}
v_{dq0s} &= -K_s R_s K_s^{-1} i_{dq0s} + K_s \tfrac{d}{dt}\left[K_s^{-1}\right]\lambda_{dq0s} + \tfrac{d}{dt}[\lambda_{dq0s}], \\
v'_{dq0r} &= K_r R'_r K_r^{-1} i'_{dq0r} + K_r \tfrac{d}{dt}\left[k_r^{-1}\right]\lambda'_{dq0r} + \tfrac{d}{dt}[\lambda'_{dq0r}], \\
\lambda_{dq0s} &= -K_s L_{ss} K_s^{-1} i_{dq0s} + K_s L'_{sr} K_r^{-1} i'_{dq0r}, \\
\lambda'_{dq0r} &= -K_r L'^{T}_{sr} K_s^{-1} i_{dq0s} + K_r L'_{rr} K_r^{-1} i'_{dq0r},
\end{aligned} \tag{B.26}$$

where

$$K_s R_s K_s^{-1} = R_s,$$
$$K_r R'_r K_r^{-1} = R'_r,$$
$$K_s \frac{d}{dt}\left[K_s^{-1}\right] = \begin{bmatrix} 0 & -\omega_s & 0 \\ \omega_s & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$
$$K_r \frac{d}{dt}\left[K_r^{-1}\right] = \begin{bmatrix} 0 & -(\omega_s - \omega_r) & 0 \\ (\omega_s - \omega_r) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$
$$K_s L_{ss} K_s^{-1} = \begin{bmatrix} L_s & 0 & 0 \\ 0 & L_s & 0 \\ 0 & 0 & L_{ls} \end{bmatrix},$$
$$K_s L'_{sr} K_r^{-1} = \begin{bmatrix} L_m & 0 & 0 \\ 0 & L_m & 0 \\ 0 & 0 & 0 \end{bmatrix},$$
$$K_r L'_{rr} K_r^{-1} = \begin{bmatrix} L_r & 0 & 0 \\ 0 & L_r & 0 \\ 0 & 0 & L'_{ls} \end{bmatrix},$$

with

$$L_m = \tfrac{3}{2} L_{ms},$$
$$L_s = L_{ls} + L_m,$$
$$L_r = L'_{ls} + L_m.$$

B.1.2 TORQUE EQUATION IN ARBITRARY REFERENCE-FRAME VARIABLES

The torque equation in the three phase variables of the DFIG is defined as [38]

$$\tau_e = \left(\frac{P}{2}\right) (i_{abcs})^T \frac{\partial}{\partial \theta_r} [L'_{sr}]\, i'_{abcr}, \tag{B.27}$$

where P is a pair of poles. The term $\frac{\partial}{\partial \theta_r}[L'_{sr}]$ in (B.27) is defined as

$$\frac{\partial}{\partial \theta_r}[L'_{sr}] = \begin{bmatrix} -L_{ms}\sin\theta_r & -L_{ms}\sin\left(\theta_r + \frac{2\pi}{3}\right) & -L_{ms}\sin\left(\theta_r - \frac{2\pi}{3}\right) \\ -L_{ms}\sin\left(\theta_r - \frac{2\pi}{3}\right) & -L_{ms}\sin\theta_r & -L_{ms}\sin\left(\theta_r + \frac{2\pi}{3}\right) \\ -L_{ms}\sin\left(\theta_r + \frac{2\pi}{3}\right) & -L_{ms}\sin\left(\theta_r - \frac{2\pi}{3}\right) & -L_{ms}\sin\theta_r \end{bmatrix}. \tag{B.28}$$

The expression for the electromagnetic torque in terms of arbitrary reference-frame variables may be obtained by substituting the equations of transformation into (B.27). Thus [38]

$$\tau_e = \left(\frac{P}{2}\right)\left(K_s^{-1} i_{dq0s}\right)^T \frac{\partial}{\partial \theta_r}\left[L'_{sr}\right]\left(K_r^{-1} i'_{dq0r}\right). \tag{B.29}$$

This expression yields the torque expressed in terms of currents as

$$\tau_e = \left(\frac{3}{2}\right)\left(\frac{P}{2}\right) L_m \left(i_{qs} i'_{dr} - i_{ds} i'_{qr}\right). \tag{B.30}$$

The torque and rotor speed in generator mode are related by

$$\frac{d\omega_r}{dt} = \dot{\omega}_r = \left(\frac{P}{2J}\right)\left(\tau_m - \tau_e\right), \tag{B.31}$$

where τ_m is the mechanical drive torque that the turbine applies to the DFIG, and J is the inertia coefficient.

B.1.3 PER-UNIT CONVERSION.

The machine and power system parameters are nearly always given in ohms or percent or per-unit of a base impedance. It is convenient to express the voltage and flux linkage equations in terms of reactances rather than inductances. Hence, B.26 is often written as [38]

$$\begin{aligned}
v_{dq0s} &= -K_s R_s K_s^{-1} i_{dq0s} + \tfrac{1}{\omega_b} K_s \tfrac{d}{dt}\left[K_s^{-1}\right] \psi_{dq0s} + \tfrac{d}{dt}\left[\psi_{dq0s}\right], \\
v'_{dq0r} &= K_r R'_r K_r^{-1} i'_{dq0r} + \tfrac{1}{\omega_b} K_r \tfrac{d}{dt}\left[K_r^{-1}\right] \psi'_{dq0r} + \tfrac{d}{dt}\left[\psi'_{dq0r}\right], \\
\psi_{dq0s} &= -\tfrac{1}{\omega_b} K_s L_{ss} K_s^{-1} i_{dq0s} + \tfrac{1}{\omega_b} K_s L'_{sr} K_r^{-1} i'_{dq0r}, \\
\psi'_{dq0r} &= -\tfrac{1}{\omega_b} K_r L'^{T}_{sr} K_s^{-1} i_{dq0s} + \tfrac{1}{\omega_b} K_r L'_{rr} K_r^{-1} i'_{dq0r},
\end{aligned} \tag{B.32}$$

where

$$\frac{1}{\omega_b} K_s \frac{d}{dt}\left[K_s^{-1}\right] = \begin{bmatrix} 0 & -\frac{\omega_s}{\omega_b} & 0 \\ \frac{\omega_s}{\omega_b} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\frac{1}{\omega_b} K_r \frac{d}{dt}\left[K_r^{-1}\right] = \begin{bmatrix} 0 & -\frac{(\omega_s-\omega_r)}{\omega_b} & 0 \\ \frac{(\omega_s-\omega_r)}{\omega_b} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\frac{1}{\omega_b} K_s L_{ss} K_s^{-1} = \begin{bmatrix} X_s & 0 & 0 \\ 0 & X_s & 0 \\ 0 & 0 & \frac{L_{ls}}{\omega_b} \end{bmatrix},$$

$$\frac{1}{\omega_b} K_s L'_{sr} K_r^{-1} = \begin{bmatrix} X_m & 0 & 0 \\ 0 & X_m & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\frac{1}{\omega_b} K_r L'_{rr} K_r^{-1} = \begin{bmatrix} X_r & 0 & 0 \\ 0 & X_r & 0 \\ 0 & 0 & \frac{L'_{ls}}{\omega_b} \end{bmatrix},$$

with ω_b as the base electrical angular velocity used to calculate the inductive reactances, as follows:

$$\begin{aligned} \psi_x &= \frac{\lambda_x}{\omega_b}, \\ X_m &= \frac{L_m}{\omega_b}, \\ X_s &= \frac{L_s}{\omega_b}, \\ X_r &= \frac{L_r}{\omega_b}. \end{aligned}$$

It is often convenient to express the machine parameters and variables as per-unit quantities. Base power (P_b) and base voltage (V_b) are selected, and all parameters and variables are normalized using these base quantities [38]. The base power may be expressed as

$$P_b = \frac{3}{2} V_b I_b. \tag{B.33}$$

Therefore, because base voltage (V_b) and base power (P_b) are selected, the base current can be calculated from (B.33). It follows that the base impedance may be expressed as

$$Z_b = \frac{V_b}{I_b} = \frac{3V_b^2}{2P_b}. \tag{B.34}$$

The $dq0$ equations written in terms of reactances, (B.32), can be readily converted to per-unit by dividing the voltages by V_b, the currents by I_b, and the resistances and reactances by Z_b [38].

Although the voltage and flux linkage per second equations do not change form when per-unitized, the torque equation is modified by the per-unitizing process. For this purpose the base torque may be expressed as [38]

$$\tau_b = \frac{P_b}{(2/P)\,\omega_b}. \tag{B.35}$$

The system (B.32) is rewritten in per-unit as follows:

$$\begin{aligned}
v_{dq0s(pu)} &= -K_s R_s K_{s(pu)}^{-1} i_{dq0s(pu)} + K_s \frac{d}{dt}\left[K_s^{-1}\right]_{(pu)} \psi_{dq0s(pu)} + \frac{1}{\omega_b}\frac{d}{dt}\left[\psi_{dq0s(pu)}\right], \\
v'_{dq0r(pu)} &= K_r R'_r K_{r(pu)}^{-1} i'_{dq0r(pu)} + K_r \frac{d}{dt}\left[K_r^{-1}\right]_{(pu)} \psi'_{dq0r(pu)} + \frac{1}{\omega_b}\frac{d}{dt}\left[\psi'_{dq0r(pu)}\right], \\
\psi_{dq0s(pu)} &= -K_s L_{ss} K_{s(pu)}^{-1} i_{dq0s(pu)} + K_s L'_{sr} K_{r(pu)}^{-1} i'_{dq0r(pu)}, \\
\psi'_{dq0r(pu)} &= -K_r L'^{T}_{sr} K_{s(pu)}^{-1} i_{dq0s(pu)} + K_r L'_{rr} K_{r(pu)}^{-1} i'_{dq0r(pu)},
\end{aligned} \tag{B.36}$$

where

$$K_s \frac{d}{dt}\left[K_s^{-1}\right]_{(pu)} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$K_r \frac{d}{dt}\left[K_r^{-1}\right]_{(pu)} = \begin{bmatrix} 0 & -\left(1-\omega_{r(pu)}\right) & 0 \\ \left(1-\omega_{r(pu)}\right) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$K_s L_{ss} K_{s(pu)}^{-1} = \begin{bmatrix} X_{s(pu)} & 0 & 0 \\ 0 & X_{s(pu)} & 0 \\ 0 & 0 & \frac{L_{ls}}{Z_b \omega_b} \end{bmatrix},$$

$$K_s L'_{sr} K_{r(pu)}^{-1} = \begin{bmatrix} X_{m(pu)} & 0 & 0 \\ 0 & X_{m(pu)} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$K_r L'_{rr} K_{r(pu)}^{-1} = \begin{bmatrix} X_{r(pu)} & 0 & 0 \\ 0 & X_{r(pu)} & 0 \\ 0 & 0 & \frac{L'_{ls}}{Z_b \omega_b} \end{bmatrix},$$

where $\omega_b = \omega_s$.

The electric torque τ_e (B.30) and the rotor speed equation (B.31) are rewritten in *pu* as follows:

$$\tau_{e(pu)} = X_{m(pu)} \left(i_{qs(pu)} i'_{dr(pu)} - i_{ds(pu)} i'_{qr(pu)}\right). \tag{B.37}$$

$$\frac{d\omega_{r(pu)}}{dt} = \dot{\omega}_{r(pu)} = \left(\frac{1}{2H}\right)\left(\tau_{m(pu)} - \tau_{e(pu)}\right). \tag{B.38}$$

Note: *To facilitate the calculations, we will omit writing the subscript (pu) in all variables used in this book hereafter; the read can take for granted that they are in (pu) unless otherwise stated.*

B.1.4 DFIG STATE VARIABLES MODEL

In this book is used a DFIG. It is essentially an induction machine with wound rotor and variable frequency excitation by the rotor circuit, which is controlled by means of power converters. DFIG configuration allows the rotor speed to vary while synchronizing the stator directly to a fixed frequency power system, the control input is by the rotor winding, and in practice it is possible to measure all the DFIG currents. So, it is convenient to select the stator and rotor currents like DFIG state variables.

Then, in (B.36), substituting ψ_{dq0s} and $\boldsymbol{\psi'}_{\boldsymbol{dq0r}}$ in the v_{dq0s} and $\boldsymbol{v'}_{\boldsymbol{dq0r}}$ equations, respectively,

$$\begin{aligned} v_{dq0s} &= -K_s R_s K_s^{-1} i_{dq0s} + K_s \frac{d}{dt}\left[K_s^{-1}\right] \left(-K_s L_{ss} K_s^{-1} i_{dq0s} + K_s L'_{sr} K_r^{-1} i'_{dq0r}\right) \\ &\quad + \frac{d}{dt}\left[-K_s L_{ss} K_s^{-1} i_{dq0s} + K_s L'_{sr} K_r^{-1} i'_{dq0r}\right], \end{aligned} \tag{B.39}$$

$$\begin{aligned} v'_{dq0r} &= K_r R'_r K_r^{-1} i'_{dq0r} + K_r \frac{d}{dt}\left[K_r^{-1}\right] \left(-K_r L'^{T}_{sr} K_s^{-1} i_{dq0s} + K_r L'_{rr} K_r^{-1} i'_{dq0r}\right) \\ &\quad + \frac{d}{dt}\left[-K_r L'^{T}_{sr} K_s^{-1} i_{dq0s} + K_r L'_{rr} K_r^{-1} i'_{dq0r}\right]. \end{aligned} \tag{B.40}$$

Equations (B.39) and (B.40) can be rewritten as

$$\begin{aligned} v_{dqos} &= -\left(K_s R_s K_s^{-1} + K_s \frac{d}{dt}\left[K_s^{-1}\right] K_s L_{ss} K_s^{-1}\right) i_{dqos} + K_s \frac{d}{dt}\left[K_s^{-1}\right] K_s L'_{sr} K_r^{-1} i'_{dq0r} \\ &\quad - K_s L_{ss} K_s^{-1} \frac{d}{dt}[i_{dqos}] + K_s L'_{sr} K_r^{-1} \frac{d}{dt}[i'_{dqor}], \end{aligned} \tag{B.41}$$

$$\begin{aligned} v'_{dq0r} &= -K_r \frac{d}{dt}\left[K_r^{-1}\right] K_r L'^{T}_{sr} K_s^{-1} i_{dqos} + \left(K_r R'_r K_r^{-1} + K_r \frac{d}{dt}\left[K_r^{-1}\right] K_r L'_{rr} K_r^{-1}\right) i'_{dq0r} \\ &\quad - K_r L'^{T}_{sr} K_s^{-1} \frac{d}{dt}\left[i_{dqos}\right] + K_r L'_{rr} K_r^{-1} \frac{d}{dt}[i'_{dqor}]. \end{aligned} \tag{B.42}$$

In order to simplify the handling equations, equations (B.41) and (B.42) can be rewritten in the matrix form as

$$\begin{aligned} &\begin{bmatrix} v_{dq0s} \\ v'_{dq0r} \end{bmatrix} \\ &= \begin{bmatrix} -K_s R_s K_s^{-1} - K_s \frac{d}{dt}\left[K_s^{-1}\right] K_s L_{ss} K_s^{-1} & K_s \frac{d}{dt}\left[K_s^{-1}\right] K_s L'_{sr} K_r^{-1} \\ -K_r \frac{d}{dt}\left[K_r^{-1}\right] K_r L'^{T}_{sr} K_s^{-1} & K_r R'_r K_r^{-1} + K_r \frac{d}{dt}\left[K_r^{-1}\right] K_r L'_{rr} K_r^{-1} \end{bmatrix} \\ &\quad \begin{bmatrix} i_{dq0s} \\ i'_{dq0r} \end{bmatrix} + \begin{bmatrix} -K_s L_{ss} K_s^{-1} & K_s L'_{sr} K_r^{-1} \\ -K_r L'^{T}_{sr} K_s^{-1} & K_r L'_{rr} K_r^{-1} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_{dq0s} \\ i'_{dq0r} \end{bmatrix}, \end{aligned} \tag{B.43}$$

where each matrix term is defined as

$$-K_s R_s K_s^{-1} - K_s \frac{d}{dt}\left[K_s^{-1}\right] K_s L_{ss} K_s^{-1} = \begin{bmatrix} -r_s & X_s & 0 \\ -X_s & -r_s & 0 \\ 0 & 0 & -r_s \end{bmatrix}, \tag{B.44}$$

$$K_s \frac{d}{dt}\left[K_s^{-1}\right] K_s L'_{sr} K_r^{-1} = \begin{bmatrix} 0 & -X_m & 0 \\ X_m & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{B.45}$$

$$-K_r \frac{d}{dt}\left[K_r^{-1}\right] K_r L'^{T}_{sr} K_s^{-1} = \begin{bmatrix} 0 & -X_m(\omega_r - 1) & 0 \\ X_m(\omega_r - 1) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{B.46}$$

$$K_r R'_r K_r^{-1} + K_r \frac{d}{dt}\left[K_r^{-1}\right] K_r L'_{rr} K_r^{-1} = \begin{bmatrix} r'_r & X_r(\omega_r - 1) & 0 \\ -X_r(\omega_r - 1) & r'_r & 0 \\ 0 & 0 & r'_r \end{bmatrix}. \tag{B.47}$$

Then equation (B.43) can be rewritten as

$$v_{dq0} = Z i_{dq0} + L \frac{d}{dt} i_{dq0}, \tag{B.48}$$

where

$$v_{dq0} = \begin{bmatrix} v_{ds} \\ v_{qs} \\ v_{0s} \\ v'_{dr} \\ v'_{qr} \\ v'_{0r} \end{bmatrix}, \; i_{dq0} = \begin{bmatrix} i_{ds} \\ i_{qs} \\ i_{0s} \\ i'_{dr} \\ i'_{qr} \\ i'_{0r} \end{bmatrix}, \tag{B.49}$$

$$Z = \begin{bmatrix} -r_s & X_s & 0 & 0 & -X_m & 0 \\ -X_s & -r_s & 0 & X_m & 0 & 0 \\ 0 & 0 & -r_s & 0 & 0 & 0 \\ 0 & -X_m(\omega_r - 1) & 0 & r'_r & X_r(\omega_r - 1) & 0 \\ X_m(\omega_r - 1) & 0 & 0 & -X_r(\omega_r - 1) & r'_r & 0 \\ 0 & 0 & 0 & 0 & 0 & r'_r \end{bmatrix}, \tag{B.50}$$

$$L = \begin{bmatrix} -\frac{X_s}{\omega_b} & 0 & 0 & \frac{X_m}{\omega_b} & 0 & 0 \\ 0 & -\frac{X_s}{\omega_b} & 0 & 0 & \frac{X_m}{\omega_b} & 0 \\ 0 & 0 & -\frac{L_{ls}}{\omega_b} & 0 & 0 & 0 \\ -\frac{X_m}{\omega_b} & 0 & 0 & \frac{X_r}{\omega_b} & 0 & 0 \\ 0 & -\frac{X_m}{\omega_b} & 0 & 0 & \frac{X_r}{\omega_b} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{L'_{ls}}{\omega_b} \end{bmatrix}. \tag{B.51}$$

In (B.48), it is ease to solve $\frac{d}{dt} i_{dq0}$ as follows:

$$\frac{d}{dt} I_{dq0} = -L^{-1} Z I_{dq0} + L^{-1} V_{dq0}. \tag{B.52}$$

The main feature of the transformation to a frame fixed in the rotor ($d - q$) is that the variables that belong to the 0 axis are independent of ω; then these are not associated with the transformation frame. Additionally, the variables i_{0_s}, i_{0_r}, v_{0_s}, and v_{0_r} are 0 for a balanced system. Then, the system (B.52) can be reduced to

$$\frac{d}{dt} i_{dq} = A(\omega_r) i_{dq} + B v_{dq}, \tag{B.53}$$

where

$$v_{dq} = \begin{bmatrix} v_{ds} \\ v_{qs} \\ v'_{dr} \\ v'_{qr} \end{bmatrix}, \; i_{dq} = \begin{bmatrix} i_{ds} \\ i_{qs} \\ i'_{dr} \\ i'_{qr} \end{bmatrix},$$

$$A(\omega_r) = \begin{bmatrix} -\frac{\omega_b r_s}{X_s \sigma} & \omega_b \left(1 - \frac{\sigma-1}{\sigma} \omega_r\right) & -\frac{\omega_b X_m r'_r}{X_s X_r \sigma} & -\frac{\omega_b X_m}{X_s \sigma} \omega_r \\ -\omega_b \left(1 - \frac{\sigma-1}{\sigma} \omega_r\right) & -\frac{\omega_b r_s}{X_s \sigma} & \frac{\omega_b X_m}{X_s \sigma} \omega_r & -\frac{\omega_b X_m r'_r}{X_s X_r \sigma} \\ -\frac{\omega_b X_m r_s}{X_s X_r \sigma} & \frac{\omega_b X_m}{X_r \sigma} \omega_r & -\frac{\omega_b r'_r}{X_r \sigma} & \omega_b \left(1 - \frac{1}{\sigma} \omega_r\right) \\ -\frac{\omega_b X_m}{X_r \sigma} \omega_r & -\frac{\omega_b X_m r_s}{X_s X_r \sigma} & -\omega_b \left(1 - \frac{1}{\sigma} \omega_r\right) & -\frac{\omega_b r'_r}{X_r \sigma} \end{bmatrix},$$

$$B = \begin{bmatrix} \frac{-\omega_b}{X_s \sigma} & 0 & \frac{\omega_b X_m}{X_s X_r \sigma} & 0 \\ 0 & \frac{-\omega_b}{X_s \sigma} & 0 & \frac{\omega_b X_m}{X_s X_r \sigma} \\ -\frac{\omega_b X_m}{X_s X_r \sigma} & 0 & \frac{\omega_b}{X_r \sigma} & 0 \\ 0 & -\frac{\omega_b X_m}{X_s X_r \sigma} & 0 & \frac{\omega_b}{X_r \sigma} \end{bmatrix},$$

with

$$\sigma = 1 - \frac{X_m^2}{X_s X_r}.$$

The equations (B.38) and (B.53) are the state space representation of the DFIG. This representation has four electrical variables (i_{ds}, i_{qs}, i_{dr}, i_{qr}) and one mechanical variable

(ω_r).

B.2 DC LINK MATHEMATICAL MODEL

The DFIG allows the rotor speed to vary while synchronizing the stator directly to a fixed frequency power system. This is achieved by controlling the rotor side converter (RSC). The RSC is connected via a DC Link to a grid side converter (GSC), which is in turn connected to the stator terminals directly or through a step-up transformer. The circuit of the DC Link connected to the GSC and the last one connected to the stator terminals can be considered as a STATCOM, as shown in Figure B.4. The GSC block in the circuit is treated as an ideal. Assuming balanced conditions, the AC-side circuit equations in Figure B.4 can be written as

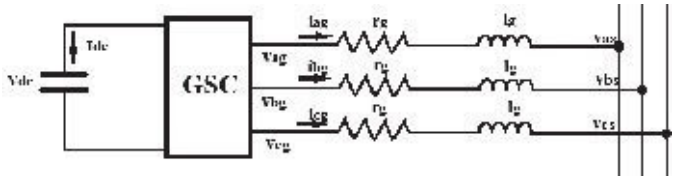

Figure B.4 DC Link configuration (STATCOM).

$$\begin{aligned} v_{ag} - v_{as} &= r_g i_{ag} + \frac{d\lambda_{ag}}{dt}, \\ v_{bg} - v_{bs} &= r_g i_{bg} + \frac{d\lambda_{bg}}{dt}, \\ v_{cg} - v_{cs} &= r_g i_{cg} = \frac{d\lambda_{cg}}{dt}, \end{aligned} \tag{B.54}$$

$$\begin{aligned} \lambda_{ag} &= l_g i_{ag}, \\ \lambda_{bg} &= l_g i_{bg}, \\ \lambda_{ag} &= l_g i_{ag}, \end{aligned} \tag{B.55}$$

where i_{ag}, i_{bg}, and i_{cg} are the currents in the *abc* axis, respectively; v_{as}, v_{bs}, and v_{cs} are the three-phase grid voltages; v_{ag}, v_{bg}, and v_{cg} are the three-phase voltages generated by the GSC, they are the control input for the DC Link circuit. Equations (B.54) and (B.55) can be rewritten in the matrix form as follows:

$$v_{abdcg} - v_{abcs} = R_g i_{abcg} + \frac{d\lambda_{abcg}}{dt}, \tag{B.56}$$

$$\lambda_{abcg} = L_g i_{abcg}, \tag{B.57}$$

where

$$v_{abcg} = \begin{bmatrix} v_{ag} \\ v_{bg} \\ v_{cg} \end{bmatrix}, v_{abcs} = \begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} i_{abcg} = \begin{bmatrix} i_{ag} \\ i_{bg} \\ i_{cg} \end{bmatrix}, \lambda_{abcg} = \begin{bmatrix} \lambda_{ag} \\ \lambda_{bg} \\ \lambda_{cg} \end{bmatrix},$$

$$R_g = \begin{bmatrix} r_g & 0 & 0 \\ 0 & r_g & 0 \\ 0 & 0 & r_g \end{bmatrix}, L_g = \begin{bmatrix} l_g & 0 & 0 \\ 0 & l_g & 0 \\ 0 & 0 & l_g \end{bmatrix}.$$

By the same reasons described in Subsection B.1.1, the system (B.56) and (B.2) can be written in a synchronously rotating reference frame using the *d*-*q* transformation. Then the transformation used is defined as

$$K_s = \frac{2}{3} \begin{bmatrix} \cos\theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ -\sin\theta & -\sin\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \tag{B.58}$$

$$K_s^{-1} = \begin{bmatrix} \cos\theta & -\sin\theta & 1 \\ \cos\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\theta + \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) & 1 \end{bmatrix}. \tag{B.59}$$

Now, applying the transformation to equations (B.56), and (B.2) the follow equation is obtained:

$$K_s^{-1} v_{dq0g} - K_s^{-1} v_{dq0s} = R_g K_s^{-1} i_{dq0g} + \frac{d}{dt}\left(K_s^{-1} \lambda_{dq0g}\right)$$
$$K_s^{-1} \lambda_{dq0g} = L_g K_s^{-1} i_{dq0g}$$

and, finally, the change of variable is obtained as

$$v_{dq0g} - v_{dq0s} = K_s R_g K_s^{-1} i_{dq0g} + K_s \frac{d}{dt}\left(K_s^{-1}\right) \lambda_{dq0g} + \frac{d}{dt}\left(\lambda_{dq0g}\right), \tag{B.60}$$

$$\lambda_{dq0g} = K_s L_g K_s^{-1} i_{dq0g}, \tag{B.61}$$

where

$$K_s R_g K_s^{-1} = R_g = \begin{bmatrix} r_g & 0 & 0 \\ 0 & r_g & 0 \\ 0 & 0 & r_g \end{bmatrix},$$

$$K_s \frac{d}{dt}\left[K_s^{-1}\right] = \begin{bmatrix} 0 & -\omega & 0 \\ \omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$K_s L_g K_s^{-1} = L_g = \begin{bmatrix} l_g & 0 & 0 \\ 0 & l_g & 0 \\ 0 & 0 & l_g \end{bmatrix}.$$

Substitute (B.61) in (B.60) as follows:

$$v_{dq0g} - v_{dq0s} = R_g i_{dq0g} + K_s \frac{d}{dt}\left(K_s^{-1}\right) L_g i_{dq0g} + \frac{d}{dt}\left(L_g i_{dq0g}\right). \tag{B.62}$$

Now, in (B.62) solving for $\frac{di_{dqg}}{dt}$, the following equation is obtained:

$$\frac{di_{dqg}}{dt} = L_g^{-1}\left(-R_g i_{dq0g} - K_s \frac{d}{dt}\left(K_s^{-1}\right) L_g i_{dq0g} + v_{dq0g} - v_{dq0s}\right). \tag{B.63}$$

As in Subsection B.1.1, the transformed variables to belong to the 0 axis are 0 for a balanced system. Then, the equation obtained above can be written as

$$\frac{di_{dqg}}{dt} = A_g i_{dqg} + B_g v_{dqg} - B_g v_{dqs}, \tag{B.64}$$

where

$$\begin{aligned} A_g &= -L_g^{-1} R_g - L_g^{-1} K_s \tfrac{d}{dt}\left(K_s^{-1}\right) L_g \\ &= \begin{bmatrix} -\frac{r_g}{l_g} & \omega \\ -\omega & -\frac{r_g}{l_g} \end{bmatrix}, \end{aligned} \tag{B.65}$$

$$\begin{aligned} B_g &= L_g^{-1} \\ &= \begin{bmatrix} \frac{1}{l_g} & 0 \\ 0 & \frac{1}{l_g} \end{bmatrix}, \\ i_{dqg} &= \begin{bmatrix} i_{dg} \\ i_{qg} \end{bmatrix}, v_{dqg} = \begin{bmatrix} v_{dg} \\ v_{qg} \end{bmatrix}, v_{dqs} = \begin{bmatrix} v_{ds} \\ v_{qs} \end{bmatrix}, \end{aligned} \tag{B.66}$$

where (i_{dg}, i_{qg}), (v_{dg}, v_{qg}), and (v_{ds}, v_{qs}) are the d-q components of (i_{ag}, i_{bg}, i_{cg}), (v_{ag}, v_{bg}, v_{cg}), and (v_{as}, v_{bs}, v_{cs}), respectively.

Negleting the harmonics due to switching and the losses in the GSC and the transformer, the power balance between the AC and DC sides of the GSC is given by

$$\frac{3}{2}\left(v_{ds}i_{dg} + v_{qs}i_{qg}\right) = v_{dc}i_{dc} = Cv_{dc}\frac{dv_{dc}}{dt}, \tag{B.67}$$

where, solving for $\frac{dv_{dc}}{dt}$ the following equation is obtained:

$$\frac{dv_{dc}}{dt} = \frac{3}{2Cv_{dc}}\left(v_{ds}i_{dg} + v_{qs}i_{qg}\right). \tag{B.68}$$

Equations (B.68) and (B.64) are the state space representation of the DC Link. This representation has three electrical variables (v_{dc}, i_{dg}, i_{qr}).

Based on Section B.1.3, the DC Link mathematical model in *pu* is obtained as

$$\frac{dv_{dc(pu)}}{dt} = \frac{1}{Cv_{dc(pu)}}v_{dqs(pu)}^{T}M_{p(pu)}v_{dqg(pu)}, \tag{B.69}$$

$$\frac{di_{dqg(pu)}}{dt} = A_{g(pu)}i_{dqg(pu)} + B_{g(pu)}v_{dqg(pu)} - B_{g(pu)}v_{dqs(pu)}, \tag{B.70}$$

where

$$M_{p(pu)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, A_{g(pu)} = \begin{bmatrix} -\frac{\omega_b r_g}{X_l} & \omega_s \\ -\omega_s & -\frac{\omega_b r_g}{X_l} \end{bmatrix}, B_{g(pu)} = \begin{bmatrix} \frac{\omega_b}{X_l} & 0 \\ 0 & \frac{\omega_b}{X_l} \end{bmatrix}$$

.

References

1. G. Abad, M. A. Rodriguez, G. Iwanski, and J. Poza. Direct power control of doubly-fed-induction-generator-based wind turbines under unbalanced grid voltage. *IEEE Transactions on Power Electronics*, 25(2):442–452, 2010.

2. Gonzalo Abad, Jesus Lopez, Miguel Rodríguez, Luis Marroyo, and Grzegorz Iwanski. *Doubly fed induction machine: modeling and control for wind energy generation*, volume 85. John Wiley & Sons, Piscataway, NJ, USA, 2011.

3. Haitham Abu-Rub, Atif Iqbal, and Jaroslaw Guzinski. *High Performance Control of AC Drives with Matlab / Simulink Models*. Wiley, West Sussex, United Kingdom, 2012.

4. R. Aghatehrani and R. Kavasseri. Sliding mode control approach for voltage regulation in microgrids with DFIG based wind generations. In 2011 IEEE Power and Energy Society General Meeting, pages 1–8, Detroit, MI, USA, July 2011.

5. A. Al-Tamimi, F.L. Lewis, and M. Abu-Khalaf. Discrete-time nonlinear HJB solution using approximate dynamic programming: Convergence proof. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 38(4):943–949, August 2008.

6. Alma Y. Alanis, Edgar N. Sanchez, Alexander G. Loukianov, and M. A. Perez-Cisneros. Real-time discrete neural block control using sliding modes for electric induction motors. *IEEE Transactions on Control Systems Technology*, 18(1):11–21, January 2010.

7. Alma Yolanda Alanis. *Neural network training using Kalman filtering*. Master's thesis, Cinvestav, Unidad Guadalajara, Guadalajara, Jalisco, México, 2004. (in spanish).

8. Giovani Luca Amicucci, Salvatore Monaco, and Dorotheé Normand-Cyrot. Control Lyapunov stabilization of affine discrete-time systems. In Proceedings of the 36th IEEE Conference on Decision and Control, volume 1, pages 923–924, San Diego, CA, USA, December 1997.

9. Olimpo Anaya-Lara, Nick Jenkins, Janaka Ekanayake, Phill Cartwright, and Mike Hughes. *Wind Energy Generation, Modelling and Control*. John Wiley & Sons, Chichester, United Kingdom, 2009.

10. World Wind Energy Association. *World wind energy report 2010*. Technical report, World Wind Energy Association, Cairo, Egypt, April 2011.

11. World Wind Energy Association. *World wind energy report 2012*. Technical report, World Wind Energy Association, Bonn, Germany, May 2013.

12. Tamer Basar and Geert Jan Olsder. *Dynamic Noncooperative Game Theory*. Academic Press, New York, USA, 2nd edition, 1995.

13. M. Bejaoui, B. Marinescu, I. Slama-Belkhodja, and E. Monmasson. Control of DFIG for wind energy in a network context: A new formulation and interpretation of the control specifications. In 2011 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe), pages 1–7, Manchester, United Kingdom, 2011.

14. B. Beltran, M. E. H. Benbouzid, and T. Ahmed-Ali. A combined high gain observer and high-order sliding mode controller for a DFIG-based wind turbine. In 2010 IEEE International Energy Conference and Exhibition (EnergyCon), pages 322–327, Manama, Bahrain, 2010.

15. BinWu, Yongqiang Lang, Navid Zargari, and Samir Kouro. *Power Conversion and Control of Wind Energy Systems*. John Wiley & Sons, Hoboken, New Jersey, USA, August 2011.

16. Ion Boldea. *Variable Speed Generators*. CRC Press, Boca Raton, FL, USA, 2005.

17. T. Burton, D. Sharpe, N. Jenkis, and E. Bossanyi. *Wind Energy Handbook*. John Wiley & Sons, New York, USA, 2001.

18. R. Cardenas, R. Pena, S. Alepuz, and G. Asher. Overview of control systems for the operation of DFIGs in wind energy applications. *IEEE Transactions on Industrial Electronics*, 60(7):2776–2798, 2013.

19. N. Cotter. The stone-weiertrass theorem and its application to neural networks. *IEEE Transactions on Neural Networks*, 1(4):290–295, 1990.

20. G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2:303–314, 1989.

21. S. V. Drakunov and V. I. Utkin. Sliding mode control in dynamic systems. *International Journal of Control*, 55(4):1029–1037, 1990.

22. R. C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In Proceedings of the Sixth International Symposium on Micromachine and Human Science, pages 39–43, Nagoya, Japan, 1995.

23. R. C. Eberhart, P. K. Simpson, and R. W. Dobbins. *Computational Intelligence PC Tools*. Academic Press Professional, Boston, MA, USA, 1996.

24. Lingling Fan and Zhixin Miao. *Modeling and Analysis of Doubly Fed Induction Generator Wind Energy Systems*. Elsevier, London, United Kingdom, 2015.

25. L. A. Feldkamp, D. V. Prokhorov, and T. M. Feldkamp. Simple and conditioned adaptive behavior from Kalman filter trained recurrent networks. *Neural Networks*, 16:683–689, 2003.

26. R. A. Felix. *Variable Structure Neural Control*. PhD thesis, Cinvestav, Unidad Guadalajara, Guadalajara, Jalisco, México, 2004.

27. Randy A. Freeman and Petar V. Kokotović. *Robust Nonlinear Control Design: State-Space and Lyapunov Techniques*. Birkhauser Boston Inc., Cambridge, MA, USA, 1996.

28. Ken-Ichi Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2(3):183–192, 1989.

29. S. S. Ge, T. H. Lee, and C. J. Harris. *Adaptive Neural Network Control for Robotic Manipulators*. World Scientific, Singapore, 1998.

30. W. A. M. Ghoneim, A. Lotfy, and K. Maher. Real time implementation of a fuzzy sensorless controller for DFIG-based variable speed wind turbine system. *European Journal of Scientific Research*, 70(2):276–291, 2012.

31. R. Grover and P. Y. C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering*. John Wiley & Sons, New York, USA, 2nd edition, 1992.

32. S. Haykin. *Kalman Filtering and Neural Networks*. John Wiley & Sons, New York, USA, 2001.

33. Kur Hornik, Maxwell Stinchcombe, and Halber White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(3):359–366, 1989.

34. Chris M. Kellett and Andrew R. Teel. Results on discrete-time control-Lyapunov functions. In Proceedings of the 42nd IEEE Conference on Decision and Control, 2003, volume 6, pages 5961–5966, Maui, Hawaii, USA, December 2003.

35. J. Kennedy and R. Eberhart. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, volume 4, pages 1942–1948, Washington, DC, USA, November 1995.

36. Donald E. Kirk. *Optimal Control Theory: An Introduction*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1970.

37. U. Kotta, S. Z. Sarpturk, and Y. Istefanopulos. Comments on the stability of discrete-time sliding mode control systems. *IEEE Transactions of Automatic Control*, 34(9):1021–1022, September 1989.

38. P. C. Krause, O. Wasyncsuk, and S. D. Sudhoff. *Analysis of Electric Machinery and Drive Sustems*. John Wiley & Sons, New York, USA, 2002.

39. P. C. Krause, O. Wasynczuk, and S. D. Sudhoff. *Analysis of Electric Machinery and Drive Systems*. John Wiley & Sons, New York, USA, 2nd edition, 1989.

40. Miroslav Krstić and Hua Deng. *Stabilization of Nonlinear Uncertain Systems*. Springer-Verlag, Berlin, Germany, 1998.

41. P. N. Lan, N. P. Quang, and P. Buechner. A non-linear control algorithm for improving performance of wind generator using doubly-fed induction generator. European Wind Energy Conference and Exhibition, Athens, Greece, May 2006.

42. Chi-Sing Leung and Lai-Wan Chan. Dual extended Kalman filtering in recurrent neural networks. *Neural Networks*, 16(2):223–239, 2003.

43. Frank L. Lewis and Vassilis L. Syrmos. *Optimal Control*. John Wiley & Sons, New York, USA, 1995.

44. Shuhui Li, M. Fairbank, C. Johnson, D. C. Wunsch, E. Alonso, and J. L. Proao. Artificial neural networks for control of a grid-connected rectifier/inverter under disturbance, dynamic and power converter switching conditions. *IEEE Transactions on Neural Networks and Learning Systems*, 25(4):738–750, April 2014.

45. Shuhui Li, Timothy A. Haskew, Yang-Ki Hong, and Ling Xu. Direct-current vector control of three-phase grid-connected rectifier-inverter. *Electric Power Systems Research*, 81(2):357–366, February 2011.

46. Chien-Hung Liu and Yuan-Yih Hsu. Effect of rotor excitation voltage on steady-state stability and maximum output power of a doubly fed induction generator. *IEEE Transactions on Industrial Electronics*, 58(4):1096–1109, 2011.

47. Alexander G. Loukianov. Robust block decomposition sliding mode control design. *Mathematical Problems in Engineering*, 8(4-5):349–365, 2002.

48. Alexander G. Loukianov and V. I. Utkin. Methods of reducing equations for dynamic systems to a regular form. *Automation and Remote Control*, 42(4):413–420, 1981.

49. Alexander Georgievich Loukianov. Nonlinear block control with sliding modes. *Automation and Remote Control*, 57(7):916–933, 1998.

50. M. Moghbel, H. T. Mokui, M. A. S. Masoum, and M. Mohseni. Reactive power control of DFIG wind power system connected to IEEE 14 bus distribution network. In Universities Power Engineering Conference (AUPEC), 2012 22nd Australasian, pages 1–7, September 2012.

51. O. A. Morfin, A. G. Loukianov, R. Ruiz, E. N. Sanchez, M. I. Castellanos, and F. A. Valenzuela. Torque controller via second order sliding modes of wrig impelled by dc-motor for application in wind systems. In American Control Conference (ACC), 2012, pages 985–990, Montreal Quebec, Canada, 2012.

52. Onofre A. Morfin, Riemann Ruiz-Cruz, Alexander G. Loukianov, Edgar N. Snchez, M.I. Castellanos, and Fredy A. Valenzuela. Torque controller of a doubly-fed induction generator impelled by a dc motor for wind system applications. *IET Renewable Power Generation*, pages 1–14, 2014.

53. K. S. Narendra and K. Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1):4–27, March 1990.

54. D. Nesic, A. R. Teel, and P. V. Kokotovic. Sufficient conditions for stabilization of sampled-data nonlinear systems via discrete-time approximations. *Systems and Control Letters*, 38(4-5):259–270, 1999.

55. Heng Nian and Yipeng Song. Direct power control of doubly fed induction generator under distorted grid voltage. *IEEE Transactions on Power Electronics*, 29(2):894–905, 2014.

56. M. Norgaard, O. Ravn, N. K. Poulsen, and L. K. Hansen. *Neural Networks for Modelling and Control of Dynamic Systems*. Springer-Verlag, New York, USA, 2000.

57. Tomoki Ohsawa, Anthony M. Bloch, and Melvin Leok. Discrete Hamilton-Jacobi theory and discrete optimal control. In Proceddings of the 49th IEEE Conference on Decision and Control (CDC), pages 5438–5443, Atlanta, GA, USA, December 2010.

58. Fernando Ornelas-Tellez. *Inverse Optimal Control for Discrete-Time Nonlinear Systems*. PhD thesis, Centro de Investigacion y de Estudios Avanzados del IPN, Guadalajara, Mexico, August 2011.

59. Fernando Ornelas-Tellez, Edgar N. Sanchez, Alexander G. Loukianov, and J. Jesus Rico. Robust inverse optimal control for discrete-time nonlinear system stabilization. *European Journal of Control*, 20(1):38–44, January 2014.

60. Konstantinos E. Parsopoulos and Michael N. Vrahatis. *Particle Swarm Optimization and Intelligence: Advances and Applications*. Information Science Reference, New York, USA, 2010.

61. R. Pena, J. C. Clare, and G. M. Asher. Doubly fed induction generator using back-to-back pwm converters and its application to variable-speed wind-energy generation. IEEE Proceedings Electric Power Applications, 143(3):231–241, May 1996.

62. R. Pena, J. C. Clare, and G. M. Asher. Doubly fed induction generator using back to back pwm converters and its application to variable speed wind energy generation. *IEEE Proceedings, Electric Power Applications*, 143(3):231–241, May 1996.

63. Van-Tung Phan and Hong-Hee Lee. Performance enhancement of stand-alone DFIG systems with control of rotor and load side converters using resonant controllers. *IEEE Transactions on Industry Applications*, 48(1):199–210, 2012.

64. A. S. Poznyak, Edgar Nelson Sanchez, and W. Yu. *Differential Neural Networks for Robust Nonlinear Control*. World Scientific, Singapore, 2001.

65. Wei Qiao, G. K. Venayagamoorthy, and R. G. Harley. Real-time implementation of a statcom on a wind farm equipped with doubly fed induction generators. *IEEE Transactions on Industry Applications*, 45(1):98–107, January 2009.

66. Wei Qiao, Wei Zhou, J. M. Aller, and R. G. Harley. Wind speed estimation based sensorless output maximization control for a wind turbine driving a DFIG. *IEEE Transactions on Power Electronics*, 23(3):1156–1169, 2008.

67. Liu Qizhong, Yu Lan, and Wu Guoxiang. Comparison of control strategy for doublefed induction generator (DFIG). In 2011 Third International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), volume 1, pages 741–744, Shanghai, China, 2011.

68. G. A. Rovithakis and M. A. Chistodoulou. *Adaptive Control with Recurrent High-Order Neural Networks*. Springer-Verlag, Berlin, Germany, 2000.

69. Riemann Ruiz, Edgar Nelson Sanchez, and Alexander Georgievich Loukianov. Realtime sliding mode control for a doubly fed induction generator. In Proceedings of the IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), pages 2975–2980, Orlando, FL, USA, December 2011.

70. Riemann Ruiz, Edgar Nelson Snchez, and Alexander Georgievich Loukianov. Discrete time block control of a double fed induction generator using sliding modes. In IEEE Multi-Conference on Systems and Control (MSC), St. Petersburg, Russia, July 2009.

71. R. Ruiz-Cruz, E. N. Sanchez, F. Ornelas-Tellez, A. G. Loukianov, and R. G. Harley. Particle swarm optimization for discrete-time inverse optimal control of a doubly fed induction generator. *IEEE Transactions on Cybernetics*, 43(6):1698–1709, December 2013.

72. Edgar Nelson Sánchez, Alma Yolanda Alanis, and Alexander Georgievich Loukianov. *Discrete-Time High Order Neural Control: Trained with Kalman Fitering*. Springer Verlag, Berlín, Germany, 2008.

73. Edgar Nelson Sánchez and Fernando Ornelas-Tellez. *Discrete-Time Inverse Optimal Control for Nonlinear Systems*. Taylor & Francis, Boca Raton, FL, USA, 2013.

74. Edgar Nelson Sanchez and Luis J. Ricalde. Trajectory tracking via adaptive recurrent neural control with input saturation. In Proceedings of International Joint Conference on Neural Networks '03, volume 4, Portland, OR, USA, July 2003.

75. Rodolphe Sepulchre, Mrdjan Jankovic, and Petar V. Kokotović. *Constructive Nonlinear Control*. Springer-Verlag, Berlin, Germany, 1997.

76. Y. Song and J. W. Grizzle. The extended Kalman filter as local asymptotic observer for discrete-time nonlinear systems. *Journal of Mathematical Systems, Estimation and Control*, 5(1):59–78, 1995.

77. A. Tapia, G. Tapia, J. X. Estolaza, and J. R. Saenz. Modeling and control of a wind turbine driven doubly fed induction generator. *IEEE Transactions on Energy Conversion*, 26(2):194–204, 2003.

78. Shaocheng Tong and Han-Xiong Li. Fuzzy adaptive sliding-mode control for mimo nonlinear systems. *IEEE Transactions on Fuzzy Systems*, 11(3):354–360, 2003.

79. V. I. Utkin. *Variable Structure and Lyapunov Control*, pages 87–107. Springer-Verlag, Berlin, Germany, 1993.

80. V. I. Utkin, J. Guldner, and J. Shi. *Sliding Mode Control in Electromechanical Systems*. Taylor & Francis, Philadelphia, PA, USA, 1999.

81. A. Varga. A schur method for pole assigment. *IEEE Transactions on Automatic Control*, 26(2):517–519, 1981.

82. G. K. Venayagamoorthy, R. G. Harley, and D. C. Wunsch. Comparison of heuristic dynamic programming and dual heuristic programming adaptive critics for neurocontrol of a turbogenerator. *IEEE Transactions on Neural Networks*, 13(3):764–773, May 2002.

83. Vijay Vittal and Raja Ayyanar. *Grid Integration and Dynamic Impact of Wind Energy*. Springer, New York, USA, 2012.

84. R. J. Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1:270–280, 1989.

85. Xinghuo Yu and O. Kaynak. Sliding-mode control with soft computing: A survey. *IEEE Transactions on Industrial*

Electronics, 56(9):3275–3285, 2009.

86. Huaguang Zhang, Ruizhuo Song, Qinglai Wei, and Tieyan Zhang. Optimal tracking control for a class of nonlinear discrete-time systems with time delays based on heuristic dynamic programming. *IEEE Transactions on Neural Networks*, 22(12):1851–1862, December 2011.

87. Xuemei Zheng, Wei Li, and Wei Wang. High-order sliding mode controller for no-load cutting-in control in DFIG wind power system. In 2010 3rd International Symposium on Systems and Control in Aeronautics and Astronautics (ISSCAA), pages 1304–1308, Harbin, China, 2010.

88. Dawei Zhi, Lie Xu, and B. W. Williams. Model-based predictive direct power control of doubly fed induction generators. *IEEE Transactions on Power Electronics*, 25(2):341–351, 2010.

Index