2013

MULTIOBJETIVE FINITE CONTROL SET MODEL PREDICTIVE TORQUE AND STATOR FLUX CONTROL OF AN INDUCTION MACHINE

ROJAS MONRROY, CHRISTIAN ALEXIS

Universidad Técnica Federico Santa María

Universidad Técnica Federico Santa María
Departamento de Electrónica

Doctoral Dissertation

Multiobjective Finite Control Set Model Predictive Torque and Stator Flux Control of an Induction Machine

Doctorate Program
Doctorate in Electronic Engineering

Candidate
Christian Alexis Rojas Monrroy

Thesis Supervisor
Dr. José Rodríguez Pérez

Evaluation Committee
Dr. César Silva Jiménez
Dr. José Espinoza Castro

July 17, 2013, Valparaíso, Chile

Remember: Your focus determines your reality...
-Qui-Gon Jinn

A La Fuerza

ACKNOWLEDGMENT

I WOULD like to express my deep gratitude, to all those who have contributed to my academic preparation and most especially, to my thesis supervisor Professor José Rodríguez. Thanks to his confidence and research support it has been possible to complete this work. I really acknowledge the valuable assistance and technical support from Professor César Silva, his example motivated me to enjoy the laboratory work. I also appreciate the support from Professors José Espinoza, Marcelo Pérez and Juan Yuz for their contributions in the development of this project.

This work was supported by FONDECYT, under grant N° 1100404, *High Performance Control of Electrical Machines*. In addition, the scholarship for the doctoral program was provided by CONICYT, under program scholarships for PhD studies in Chile and by the Universidad Técnica Federico Santa María under the Dirección General de Investigación y Postgrado.

My sincere thanks for the friendship and support of all the members of the Power Electronics Research Laboratory, PowerLab, Universidad Técnica Federico Santa María, specially Mauricio Trincado, Alan Wilson, Salvador Revelo, Hector Young, Gonzalo Carrasco, Marco Rivera, Ricardo Lizana, Ricardo Pérez, Carlos Reusser, Sebastián Rivera, Sebastián Muñoz and many others who will always be in my ROM. Thank you all for helping in the laboratory and sharing great moments during these years. I also would like acknowledge to my friend and colleague Felipe Villarroel. Thank you for your friendship and for always giving me your best feedback.

I really appreciate the fundamental support of **my family**. Thank you for enduring my absence for many years, this work belongs to you. Finally, I would like to thank my girlfriend **Cheryl**, without whose love and understanding it would not have been possible to finish this project. For all the above, thank you very much.

Christian Alexis Rojas Monrroy

ABSTRACT

DURING THE last decades, basically two control strategies for electrical drives have dominated high-performance industrial applications: field-oriented control (FOC) and direct torque control (DTC). Nowadays, these control strategies are implemented on digital platforms. Digital signal processors (DSP) allows high flexibility, the integration of more functionality, and the implementation of more complex control schemes. Due to the development of powerful and fast microprocessors, increasing attention has been dedicated to the use of model predictive control (MPC) in power electronics. The first ideas about this strategy applied to power converters started in the 1980s. The main concept is based on the calculation of the future system behavior to compute optimal actuation variables.

Due to the wide range of MPC methods, the MPC techniques applied to power electronics have been classified into two main categories: classical MPC and finite control set MPC (FCS-MPC). In the first type, the control variable is the converter output voltage, in the form of a continuous duty cycle, while an open-loop receding horizon optimization problem is solved at every sampling step to calculate the best actuation. This actuation is applied usually using pulsewidth modulation (PWM) or space vector modulation (SVM). The second type, uses the inherent discrete nature of the power converter to solve the optimization problem using a single cost function. Here, the input is restricted to a finite set of discrete values. The discrete system model is evaluated for every possible actuation sequence and then compared with the reference in order to select the best voltage vector.

The research done up to now has revealed that a key issue in FCS-MPC implementations is the selection of the weighting factors used in the cost function. Weighting factors are used to give more importance to one or another variable and to normalize the different control objectives. These scalar factors are parameters to adjust, and its selection is an important task because it is more complex than the tuning of proportional-integral (PI) coefficients or hysteresis bands of traditional controllers. Several methods using offline and online search procedures have been implemented at the present state of the art, but they are strongly dependent on the system parameters and they are formulated for two control objectives in a specific application only. When more objectives are considered, the weighting factors are usually obtained using trial and error procedures and running time-consuming simulations.

The use of a single cost function to solve the optimization problem at each sampling time is not the only possible alternative. The possibility of the use of a different optimizer is the origin of this work. Different simple multiobjective optimization methods in order to eliminate the requirement of weighting factors in the predictive torque and flux control

(PTC) scheme are presented. The optimization problem is solved using a multiobjective approach, giving rise to a multiobjective predictive torque and flux control. The scheme is then applied to an induction machine drive fed by a commercial two-level voltage source inverter (2L-VSI).

Keywords

Predictive Control, Variable Speed Drives, Optimization Methods.

RESUMEN

DURANTE LAS últimas décadas, básicamente dos estrategias de control para accionamientos eléctricos de velocidad variable han dominado la industria de aplicaciones de alto desempeño: el control orientado de flujo (FOC) y el control directo de torque (DTC). En la actualidad, estos esquemas de control son implementados comúnmente en procesadores digitales de señales (DSP). Estas plataformas permiten alta flexibilidad, integración de más periféricos y la implementación de complejos esquemas de control. Debido al desarrollo de rápidos y poderosos microprocesadores, la utilización del control predictivo basado en modelos (MPC) en electrónica de potencia ha tenido gran atención. Las primeras ideas sobre la utilización de estos métodos en accionamientos surgieron en la década de 1980. El concepto se basa en el cálculo del comportamiento futuro del sistema y sus actuaciones óptimas.

Debido a la amplia variedad de métodos MPC, su utilización en electrónica de potencia ha sido clasificado básicamente en dos categorías: MPC clásico y control predictivo con conjunto de control finito (FCS-MPC). En el primer caso, las variables de control son los voltajes del convertidor en forma de ciclos de trabajo de naturaleza continua, mientras que el problema de optimización para determinar la mejor actuación es resuelto fuera de línea. La actuación es aplicada utilizando modulación de ancho de pulso (PWM) o modulación de vectores espaciales (SVM). El segundo caso, FCS-MPC, utiliza la naturaleza discrete del convertidor de potencia para resolver el problema de optimización mediante la minimización de un funcional de costo. En este tipo de controles, la actuación es restringida a un conjunto finito de valores. Luego, el sistema discreto del modelo es evaluado para cada posible secuencia de actuación y comparado con la señal de referencia. Finalmente, el vector de voltaje que minimiza el funcional de costo en cada instante de muestreo es seleccionado.

La investigación realizada hasta ahora, ha revelado que un problema de implementación importante del esquema FCS-MPC es la selección de los factores de peso del funcional de costo. Estos factores de peso son utilizados para dar mayor importancia a una variable de control en función de otra y también para normalizar diferentes objetivos de control. Estos escalares son factores a ajustar y su selección suele ser una tarea compleja, comparada con la sintonización de un controlador lineal proporcional-integral (PI) o una banda de histéresis de esquemas tradicionales. Diversos procedimientos de búsqueda en tiempo real y fuera de línea son encontrados en la literatura, sin embargo aquellos métodos dependen de los parámetros del sistema y sólo son formulados para dos objetivos en aplicaciones específicas. Cuando más objetivos son considerados, estos factores de peso son obtenidos usualmente mediante un barrido de simulaciones.

El uso de un funcional de costo para resolver el problema de optimización en cada tiempo de muestreo no es la única alternativa. Diferentes métodos de optimización multiobjetivo a modo de eliminar los requerimientos de factores de peso en el control predictivo de torque y flujo (PTC) son presentados. El problema de optimización es resuelto utilizando un enfoque multiobjetivo, dando origen al control predictivo multiobjetivo de torque y flujo. Este esquema es aplicado en una máquina de inducción controlada por un inversor comercial fuente de voltaje de dos niveles (2L-VSI).

Palabras Claves

Control Predictivo, Accionamientos de Velocidad Variable, Métodos de Optimización.

CONTENTS

List of Figures

List of Tables

ACRONYMS

Uppercase

FOC	: field-oriented control
DTC	: direct torque control
MPC	: model predictive control
FCS-MPC	: finite control set model predictive control
PWM	: pulsewidth modulation.
SVM	: space vector modulation
PI	: proportional-integral coefficients of linear controllers
PTC	: predictive torque control
MOO	: multiobjective optimization
2L-VSI	: two-level voltage source inverter
DSP	: digital signal processor.
FS-MPC	: finite-state model predictive control
3L-NPC	: three-level neutral-point-clamped
CHB	: cascade H-bridge inverter
FCI	: flying capacitor inverter
PCC	: predictive current control
DMPC	: direct model predictive control
IM	: induction machine
OGB	: observability-gramian based
AOF	: aggregate objective function
MCDM	: multicriteria decision-making
FMCDM	: fuzzy multicriteria decision-making
FDM	: fuzzy decision-making
THD	: total harmonic distortion
NRSMD	: normalized root-mean-square deviation
PWA	: piecewise affine
QPP	: quadratic programming problem
PSC	: predictive speed control
PCC	: predictive current control
MO	: multiobjective
MRB	: multiobjective ranking-based
MPTC	: multiobjective ranking-based predictive torque control
FPTC	: fuzzy decision-making predictive torque control
FPGA	: field programmable gate array

MPCC : multiobjective ranking-based predictive current control
FPCC : fuzzy decision-making based predictive current control
HMI : human machine interface

Lowercase

dc : direct current
ac : alternate current

SYMBOLS

Matrices

$\mathbf{A}$: state matrix, $\mathbf{A} \in \mathbb{R}^{n\times n}$
$\mathbf{B}$: input matrix, $\mathbf{B} \in \mathbb{R}^{n\times p}$
$\mathbf{C}$: output matrix, $\mathbf{C} \in \mathbb{R}^{q\times n}$
$\mathbf{D}$: feedforward matrix, $\mathbf{D} \in \mathbb{R}^{q\times p}$
$\mathbf{A_d}$: discrete-time state matrix, $\mathbf{A_d} \in \mathbb{R}^{n\times n}$
$\mathbf{B_d}$: discrete-time input matrix, $\mathbf{B_d} \in \mathbb{R}^{n\times p}$
$\mathbf{C_d}$: discrete-time output matrix, $\mathbf{C_d} \in \mathbb{R}^{q\times n}$
$\mathbf{D_d}$: discrete-time feedforward matrix, $\mathbf{D_d} \in \mathbb{R}^{q\times p}$
$\mathbf{Q}$: weighting matrix of $\mathbf{y}$
$\mathbf{R}$: weighting matrix of $\mathbf{u}$
$\mathbf{P}$: weighting matrix for final values of $\mathbf{y}$
$\mathbf{I}$: identity matrix, $\mathbf{I} \in \mathbb{R}^{n\times n}$
$\mathbf{J}$: ortogonal operator matrix, $\mathbf{J} \in \mathbb{R}^{2\times 2}$
$\tilde{\mathbf{M}}$: approximation of $\mathbf{M}$, $\mathbf{M} \in \mathbb{R}^{n\times n}$
$\mathbf{T}_{abc-\alpha\beta}$: transformation matrix from abc to $\alpha\beta$
$\mathbf{T}_{\alpha\beta-dq}$: transformation matrix from $\alpha\beta$ to dq

Vectors

$\mathbf{x}$: state-variables, $\mathbf{x} = [x_1\ x_2\ \cdots\ x_n]^T$
$\mathbf{u}$: input variables, $\mathbf{u} = [u_1\ u_2\ \cdots\ u_p]^T$
$\mathbf{y}$: output variables, $\mathbf{y} = [y_1\ y_2\ \cdots\ y_q]^T$
$\mathbf{y}^*$: reference variables, $\mathbf{y}^* = [y_1^*\ y_2^*\ \cdots\ y_q^*]^T$
$\mathbf{v}_{abc}$: vector in abc-frame, $\mathbf{v} = [v_a\ v_b\ v_c]^T$
$\mathbf{v}_{\alpha\beta}$: vector in $\alpha\beta$-frame or stationary frame, $\mathbf{v} = [v_\alpha\ v_\beta]^T$
$\mathbf{v}_{dq}$: vector in dq-frame or rotating frame, $\mathbf{v} = [v_d\ v_q]^T$
$\mathbf{v_s}$: stator voltage vector
$\mathbf{i_s}$: stator current vector
$\mathbf{\Psi_s}$: stator flux vector
$\mathbf{i_r}$: rotor current vector
$\mathbf{\Psi_r}$: rotor flux vector
$\mathbf{S}$: firing pulses, $\mathbf{S} = [S_1\ S_2\ \cdots\ S_r]^T$

$\hat{\mathbf{x}}$: estimation of $\mathbf{x}$, $\hat{\mathbf{x}} = [\hat{x}_1 \ \hat{x}_2 \ \cdots \ \hat{x}_n]^T$
$\mathbf{\Delta y}$: output error, $\mathbf{\Delta y} = [\Delta y_1 \ \Delta y_2 \ \cdots \ \Delta y_q]^T$
$\mathbf{G}$: cost function vector, $\mathbf{G} = [g_1 \ g_2 \ \cdots \ g_n]^T$

Scalars

N_p : prediction horizon
N_u : receding horizon
T : electric torque
T_l : load torque
ω_f : arbitrary angular frequency
ω : rotor angular speed
ω_s : rotor flux angular speed with respect to the stator winding
G : cost function
g_i : i-th control objective
k_i : weighting factor of control objective g_i
λ_i : eigenvalues of $\mathbf{A}$, $i = 0, 1, .., dim(\mathbf{A})$

Subscripts

s : stator side
r : rotor side
a, b, c : abc-frame components
α, β : $\alpha\beta$-frame components
d, q : dq-frame components

Superscript

$*$: references
k : k-th sampling time, $\mathbf{x}(kT_s) \equiv \mathbf{x}(k) \equiv \mathbf{x}^k$
p : predicted variable, e.g., x^p
min : minimum value, e.g., $g^{\min}$
max : maximum value, e.g., $g^{\max}$
$\hat{}$: estimated value, e.g., $\hat{T}$
$\tilde{}$: average value, e.g., $\tilde{f}$

Chapter 1

INTRODUCTION

1.1 State of the Art Review

IN RECENT decades, control of electrical drives has been widely studied. Linear methods like proportional-integral (PI) controllers using pulsewidth modulation (PWM) and nonlinear methods such as hysteresis control have been fully documented in the literature and dominate high performance industrial applications [1, 2]. The most widely used linear strategy in medium and low power electrical drives is field-oriented control (FOC) [3–6], in which a decoupled torque and flux control is performed by considering an appropriate coordinate frame. A nonlinear hysteresis-based strategy such as direct torque control (DTC) appears as a solution for medium and high power applications [7].

By the end of the 1970s, model predictive control (MPC) was being used in the petrochemical industry [8–11]. The term MPC does not imply a specific control strategy, rather it covers a wide variety of control techniques that make explicit use of a mathematical model of the process and a minimization of an objective function to obtain the optimal control signals [12]. The slow dynamics of chemical processes allow long sample periods, providing enough time to solve the online optimization problem. Nowadays, the use of digital signal processors (DSP) and the development of powerful and fast microprocessors have made it possible to use MPC in the power electronics field. The continuously increasing computational power of some common hardware platforms for Power Electronics applications is shown in Fig. 1.1. The first ideas about applying MPC to power converters surfaced in the 1980s [13, 14]. The main concept is based on calculating the system's future behavior to compute optimal values for the actuating variables.

1.1.1 FSC-MPC in Power Electronics and Drives

Due to the broad range of MPC methods [15, 16], the MPC techniques applied to power electronics have been classified into two main categories: Classical MPC and finite control set MPC (FCS-MPC) or Finite-State MPC (FS-MPC) or direct MPC (DMPC) [16]. In the first type e.g., [17] and [16], the control variable is usually the converter output voltage, in the form of a duty cycle that varies continuously between its minimum and maximum magnitude, while an open-loop receding horizon optimization problem is solved at every

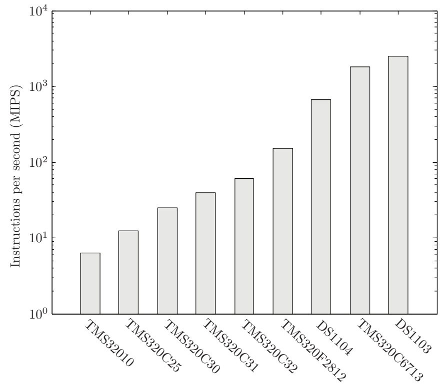


Figure 1.1: Evolution of the processing capabilities of DSP platforms commonly used in Power Electronics.

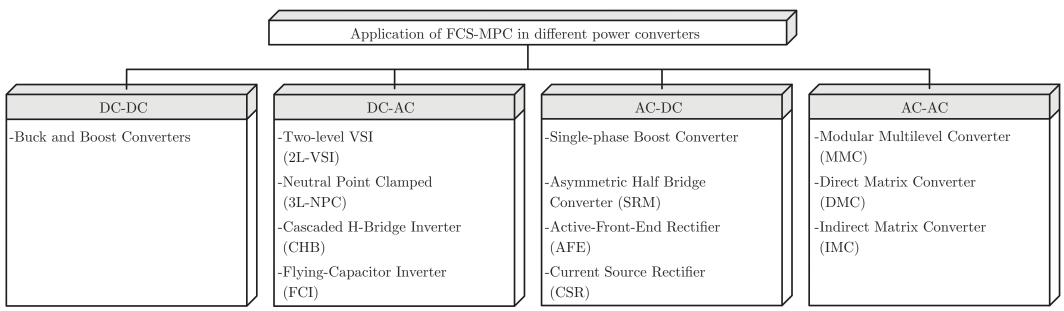


Figure 1.2: Application of FCS-MPC in different power converters.

sampling step to calculate this voltage. On the other hand, the second type, FCS-MPC, uses the inherent discrete nature of the power converter to solve the optimization problem. Here, the discrete-time model of the system is evaluated for every possible actuation sequence up to the prediction horizon N_p. Then, the outcomes of these predictions are compared to the reference to select an actuation sequence that best fits the control objectives.

Several works have reported the use of this technique on power converters such as the two-level voltage source inverter (2L-VSI) [18], three-level neutral-point-clamped (3L-NPC) [19], cascade H-bridge inverter (CHB) [20], flying capacitor inverter (FCI) [21], and matrix converters (MC) [22], whereas the use on electrical drives fed by 2L-VSI and 3L-NPC has been reported in [23–28] and [29,30] respectively. A summary of recent implementations of FCS-MPC in different power converter topologies is presented in Fig. 1.2 [31]. Each application and converter topology has its own control objectives but uses basically the same general control formulation [32].

In drive applications, FCS-MPC can be classified into two main categories according to the length of the prediction horizon: large prediction horizon $N_p \geq 2$ and short prediction horizon $N_p = 1$. An example of a large prediction horizon FCS-MPC formulation can be found in [29]. where a finite state model of a stator current control scheme is presented (PCC). In [28, 30] the same technique is used, but torque and stator flux are controlled. A comprehensive comparison between the steady state performance of short and large prediction horizon FCS-MPC with respect to FOC using PWM is presented in [33]. The main performance criteria is the compromise between switching losses and stator current (and torque) harmonic distortion achieved by each method. As expected, longer prediction horizons yield better steady state performance than horizon one. However, when larger prediction horizons or more complex converter topologies are considered, the number of calculations grows significantly. The use of only one-step prediction is a less demanding alternative in terms of computational effort and it is chosen in the current work as a benchmark to assess the transient performance of FCS-MPC method against FOC with linear controllers and PWM, [34].

In the recent years, the application of the FCS-MPC in Power Electronics has been tested and proven both theoretically and experimentally. However, the implementation of FCS-MPC in the different power converters has given rise to some questions, such as the stability of the control scheme with short and long horizons [35–37], steady-state error issues [38], weighting factors calculation and the switching frequency operation. Some of these open questions are collected in [31].

A distinctive feature of the FCS-MPC approach is the control flexibility that allows controlling current, voltage, torque, flux and other variables by designing a suitable cost function. Some of the basic control objectives that can be included in the cost function are presented in Table 1.1, [31].

1.1.2 Predictive Torque Control (PTC)

The increasing number of drive applications, in which fast dynamic response and algorithm simplicity are required, has demanded the development of new control strategies capable to

Table 1.1: Basic control objectives in FCS-MPC

Control variables	Cost functions	Nomenclature description	
Current	$g_i = \lvert i_\alpha^* - i_\alpha^p\rvert + \lvert i_\beta^* - i_\beta^p\rvert$	$i_{\alpha,\beta}^*$	: reference currents
Voltage	$g_v = \lvert v_\alpha^* - v_\alpha^p\rvert + \lvert v_\beta^* - v_\beta^p\rvert$	$v_{\alpha,\beta}^*$	: reference voltages
		$v_{\alpha,\beta}^p$	: predicted voltages
Torque	$g_T = \lvert T_e^* - T_e^p\rvert$	T_e^*	: reference torque
Flux	$g_\Psi = \lvert \lvert\lvert\mathbf{\Psi_s}\rvert\rvert^* - \lvert\lvert\mathbf{\Psi_s}\rvert\rvert^p\rvert$	$\lvert\lvert\mathbf{\Psi_s}\rvert\rvert^*$	: reference stator flux
		$\lvert\lvert\mathbf{\Psi_s}\rvert\rvert^p$	: predicted stator flux
Active	$g_P = \lvert P^* - P^p\rvert$	P^*	: reference active power
power	$P^p = v_{s\alpha} i_{s_\alpha} + v_{s\beta} i_{s\beta}$	P^p	: predicted active power
Reactive	$g_Q = \lvert Q^* - Q^p\rvert$	Q^*	: reference reactive power
power	$Q^p = v_{s\beta} i_{s_\alpha} - v_{s\alpha} i_{s\beta}$	Q^p	: predicted reactive power
Switching frequency	$g_n = n_c = \sum_i^{a,b,c} \lvert v(S_i) - v^p(S_i)\rvert$	n_c	: number of commutations to reach the next state

improve their performance. The main disadvantage in FOC is its limitation on the dynamic response [34]. It is not possible to perform it without drawbacks because of the well known cascade structure with PI-controllers, which consists in two internal current controllers and the external flux and speed controllers, which introduce a limitation in the dynamic response. Furthermore, these controllers do not have any information related to the plant, this is only required during their offline design.

In the standard PTC approach, the outer speed controller is the same as in the FOC, but the inner control loops are replaced by an one-step FCS-MPC of the stator flux and electromagnetic torque [28, 31, 34, 39]. As in any FCS-MPC, this algorithm includes a prediction of the outputs and an optimization stage. Additionally, as the stator flux is not directly measurable, it is necessary to make an estimation before the prediction, resulting in a three stage algorithm: Flux Estimation, Flux and Torque Prediction, and Cost Function Optimization. Then, the optimal voltage vector to apply in the next sampling time is selected, minimizing a simple cost function.

To obtain high performance, the weighting factors of this cost function should be selected [34]. Examining how the voltage vector is selected, the minimization of the single cost function can be recognized as a particular form of a multiobjective optimization called aggregate objective function (AOF) [40]. However, a drawback in the standard PTC scheme is that the weighting factors tuning is more complicated than that of PI coefficients or hysteresis bands used in classics controllers. To date, there is one formal method to obtain the optimal weighting factor in a cost function but only for two objectives [41]. When more objectives are considered, the weighting factors calculation is usually performed using trial and error procedures and running time-consuming simulations, in a not very systematic way [42, 43].

Another simple method to find these scalar factors was studied in [44, 45]. Although the use of the linear combined objective function to solve the optimization problem at each sampling time is simple, it is not the only possible alternative; a decision-making strategy could be used. In this way, the multiple attribute nature of the selection is retained, resulting in a multi-objective formulation and optimization [46, 47].

1.1.3 Multiobjective Optimization (MOO)

In the fiel of optimization, a multiobjective optimization (MOO) is carried out when two or more functions must be optimized simultaneously. However, to try include all the objectives in a single cost function is a complex task [8]. Over more than 40 years, many literature surveys and bibliographic reviews of MOO have been published. With the ever rapidly increasing rate of publications in the area and the development of subfields these were mostly devoted to particular aspects of multicriteria or multiobjective optimization, e.g., Multiobjective Integer Programming, Multiobjective Combinatorial Optimization, Vector Optimization, Multiobjective Evolutionary Methods, Fuzzy Multiobjective Programming, Applications of Multicriteria Decision-Making (MCDM), Goal Programming and others [40, 48]. Some of these methods are incorporated to the MPC formulation, giving rise to Multiobjective Optimal Control [49, 50].

The FCS-MPC scheme is based on the minimization of a cost function in each sampling time. The above, multiobjective optimization methods can be a good alternative to increase its flexibility. However, implementation of multiobjective approaches leads to results in an increase in the computational cost. In the traditional formulation of the FCS-MPC scheme, the controller tries to minimize the cost functions for each particular objective, minimizing an aggregate objective function composed by a linear combination of them. In [46, 47] two basic multiobjective optimization strategies are proposed to replace the aggregate cost function by a multiobjective optimization stage allowing a fair optimization of the required control objectives. The first method is based in a technique applied to the ranking of populations in evolutive optimization algorithms based on genetic algorithms [51], but it is simplified significantly since the possible solutions are already determined. The second method is based on the well-known fuzzy multicriteria decision-making (FMCDM) or fuzzy decision-making (FDM) [52].

1.2 Hypothesis and Contributions

The contribution of this research is the study and development of strategies based on FCS-MPC for the control of induction machines (IM). The main hypothesis is that a multiobjective optimization schemes can be used instead of a single cost function based on weighting factors in the conventional PTC scheme. This challenge has been faced from two different ways. The first raises the existence of an optimal weighting factor in the conventional PTC scheme, considering some merit functions or performance indices computed in one operation point. The second raises that an easy drive commissioning for predictive torque and flux control of an induction machine fed by an industrial 2L-VSI is possible by using multiobjective optimization schemes.

Two multiobjective optimization schemes are presented in this research, the first is based on a transformation of the numerical problem to an ordinal problem allowing a fair minimization of all objectives. The second method is based on well-known fuzzy multicriteria decision-making (FMCDM) to avoid the weighting factor selection. This approach has already been reported in classical MPC, but not in the context of avoiding weighting factors in the FCS-MPC scheme.

1.3 Objectives, Scope and Limitations

The general objective of this research is to avoid the selection process of weighting factors in the PTC of an induction machine. The investigation begins with the study of the problem of weighting factors selection in the PTC scheme of an induction machine. Then, the study and implementation of multiobjective optimization schemes are presented and compared with conventional schemes. Finally, a 4(kW) test rig with commercial inverters and machines is build.

The main scopes of this work are two: the two-level inverter and its components are considered ideal, where a dead-time compensation is not considered and the parameters of the induction machine are considered balanced. However, discretization of the induction machine model is studied in detail. The drive considers a cascade control loop, composed by a non-linear internal controller (torque and flux control) and with an external conventional PI-speed controller. Finally, two merit functions are considered in the problem of weighting factors selection only: total harmonic distortion (THD) and normalized root-mean-square deviation (NRSMD).

1.4 Chapter Preview

This document is divided in nine chapters. **Chapter 1** is an introduction to the subject of the research. It contains a state of the art review, establishes the contributions of the thesis and a general chapter structure. **Chapter 2** includes a general introduction to FCS-MPC.

Chapter 3 covers in detail the discretization of the induction machine model. In **Chapter 4** the conventional PTC technique and its weighting factor problem is presented. **Chapter 5** contains a review of the multiobjective theory used in MPC. In **Chapter 6** the new multiobjective PTC approaches are introduced. The experimental drive commissioning and the comparison between conventional schemes and news multiobjective PTC methods are presented in **Chapter 7**. Finally, in **Chapter 8**, the conclusions and comments from this research are included. **Appendix A** contains the list of publications in ISI Journals and international conferences derived from this research.

Chapter 2

FCS-MPC IN POWER CONVERTERS AND DRIVES

2.1 Introduction

CONVENTIONAL techniques used in static converters are based on the utilization of a modulator to generate the output voltages or currents, where the mean output values are commanded by an external signal (reference). This signal can be generated externally in an open-loop operation or given by a control scheme in closed-loop. The control scheme can be linear or non-linear. The more conventional linear controllers in drive applications are PI controllers (e.g., in FOC), while a good example of non-linear control are schemes based on dead-beat controllers. However, there are non-linear methods where the firing pulses are generated directly without any modulation stage (e.g., DTC). These techniques are called Direct Control. One of them is the FCS-MPC, which is studied in depth in this chapter.

2.2 Conventional Schemes

Standard schemes of closed-loop control used in power converters avoid the switching nature of the system. This is achieved by using a modulator, which transforms the firing pulse selection problem to a reference generation problem. The modulator applies the commanded voltage by the reference. The resultant mean voltage corresponds to the reference voltage in a carrier signal period. Then, by using a modulator it is possible to use the control theory for linear or non-linear continuous-time controllers design. Usually, these controllers are tuned under dynamic and static specifications, such as steady-state, settling-time, overshoot, bandwidth and others. The same procedure can be carried out using discrete-time controllers in digital platform implementations [1].

Another kind of standard controllers are the called Direct Controllers, in which the gate signals are generated directly without any modulator. The main difference of direct controllers with respect to conventional controllers is the avoidance of the modulation stage. The most popular direct techniques are: sliding-mode, hysteresis, DTC, [15]. The main characteristic of Power Electronics Systems is its hybrid nature. They are formed by a

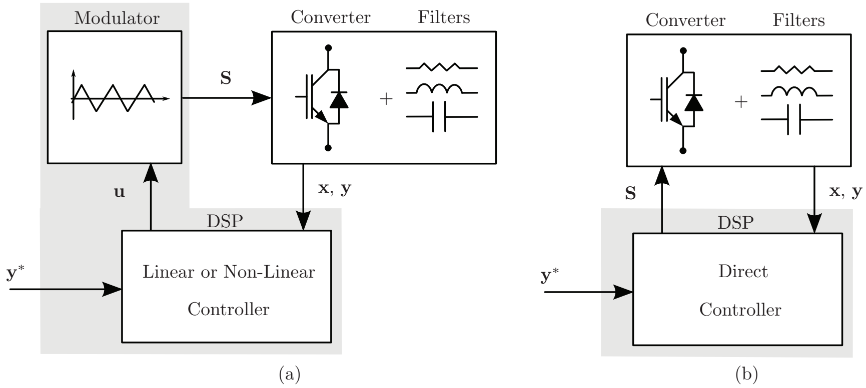


Figure 2.1: Comparison between conventional and direct controllers. (a) conventional controller; (b) direct controller.

continuous part (filters) and a discrete part (converter switches) as illustrated in Fig. 2.1. The analysis of the drive as an hybrid-systems is strongly growing in recent years [53]. A graphical comparison between conventional control are illustrated in Fig. 2.1. In the case of conventional controllers, the system information (state-variables $\mathbf{x}$ and outputs $\mathbf{y}$) is used to compute a modulation signal $\mathbf{u}$ in function to a reference $\mathbf{y}^*$. Here, the firing pulses $\mathbf{S}$ are generated by the modulator. A novel example of conventional controllers with modulator for drive applications is the well-known Field-Oriented Control (FOC). The standard control scheme of FOC is illustrated in Fig. 2.2a.

Finally, for direct controllers, firing pulses are generated directly using the system information and references. An example of direct controllers for drive applications is the well-known Direct Torque Control (DTC). The standard scheme of DTC is presented in Fig. 2.2b. The switching frequency of DTC is variable. However, by controlling the width of the tolerance bands the average switching frequency can be kept or modified.

A new recent direct controller is FCS-MPC, it uses the inherent discrete nature of the power converter to solve the optimization problem by using a single cost function minimization stage. Here, the input is restricted to a finite set of discrete values. The discrete-time system model is evaluated for every possible actuation sequence and then compared with the reference in order to select the best firing pulse sequence. A long summary of recent implementations of FCS-MPC in different power converter topologies is presented in Fig. 1.2 [31]. Each application and converter topology has its own control objectives, but it uses basically the same general control formulation [32].

2.3 Predictive Control

The main difference between Predictive Control and FOC is the precalculation of the system behavior, and its consideration in the control actuation before the difference between the reference and the measured value occurs. The feed-back PI-control loop corrects the control difference when it has already appeared.

Predictive control is a very wide class of controllers that have found rather recent application in power converters. Predictive Control is easy to understand, constraints and nonlinearities can be included, and multivariable case can be considered. This control scheme

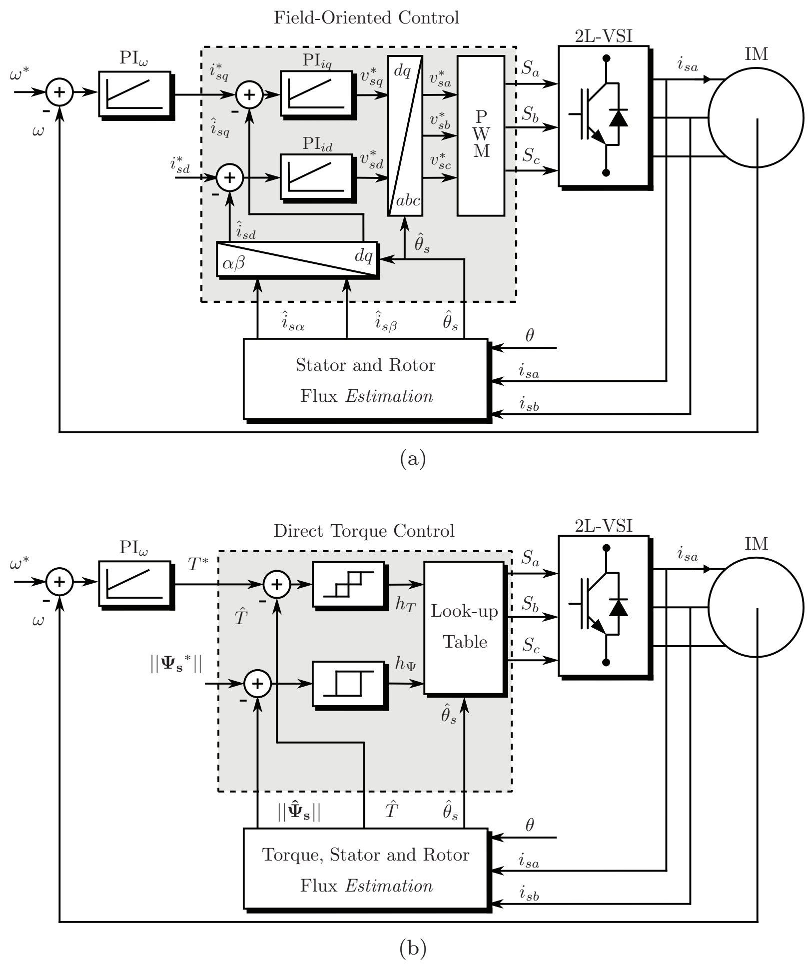


Figure 2.2: Control diagrams of conventional controllers for drive applications: (a) Field-Oriented Control; (b) Direct Torque Control.

requires a lot of calculation compared to FOC and DTC. Fortunately, the performance of modern processors have enough power processing to make this approach possible. Despite of the amount of execution steps of the control algorithm, Predictive Control has been applied in a variety of systems, demonstrated its good performance [15]. Some criteria to classify predictive controllers are presented in Fig. 2.3.

Basically, Hysteresis-, Trajectory- and Deadbeat-based predictive controllers correspond to the predictive implementations of the original concept. The current state is taken into account in order to compute the next one, e.g., in the case of hysteresis regulators, the idea is to keep the controlled variable inside a bounded area. Hence, if the variable reaches the limit, its future value is predicted for every possible value of the actuating variable and the optimum one is selected for the next sampling cycle. The same idea applies for other predictive controllers. In this case the prediction horizon is one and the integration of the inverter in the control algorithm is considered, because the firing pulses are generated immediately, without a modulator.

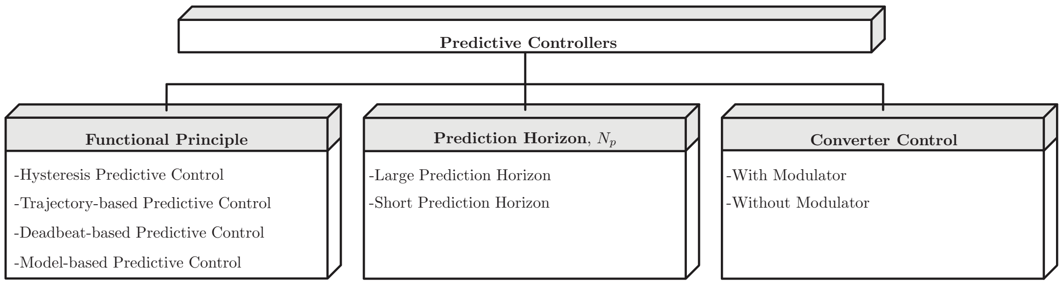


Figure 2.3: Different criteria classification of predictive controllers.

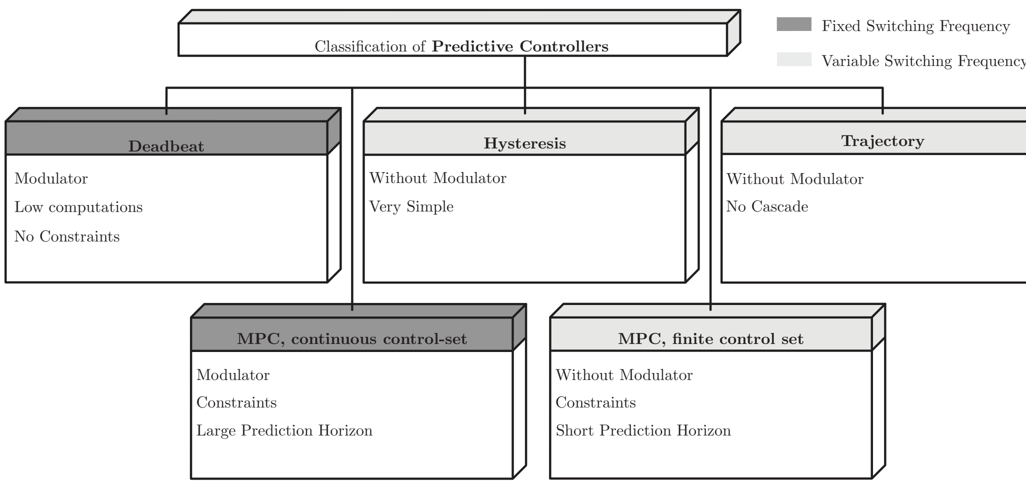


Figure 2.4: Classification of predictive control methods used in power converters.

2.4 Model Predictive Control

In Model-based Predictive Control or Model Predictive Control (MPC), the controller uses the previous and current values to predict the future behavior, it can be computed in a defined prediction horizon. The optimum switching state is selected according to the minimization of a cost function. This scheme can be implemented by considering the inverter control in the algorithm, otherwise a modulator is needed (Classical MPC or continuous control-set MPC or Explicit MPC). In Fig. 2.4 it is possible to distinguish a classification based on the functional principle of the different predictive control schemes with their main features.

2.4.1 Continuous Control-Set Model Predictive Control

The total response of the system is computed by summing the natural and forced response. This addition is calculated until the so-called prediction horizon N_p is reached. Then, the optimization is carried out by minimizing the cost function for control action variable. The selection of the structure of the cost function depends on the variables which are controlled and their references. Linear and quadratic cost functions are usually selected with the corresponding weighting factors, which may penalize the reference tracking with respect to the control effort.

In theory, MPC is able to approximate the performance over an infinite prediction horizon. Unfortunately, the constrained optimization problem needs to be solved online to find a controller output. It has computational complexity, which increases with the prediction horizon. As a consequence, the optimization horizon allows to trade off performance versus online computational effort.

The calculation of the plant behavior up to the prediction horizon N_p takes too much execution time in the control algorithm, making it impossible to perform a proper control loop. An alternative solution for this problem is to introduce a new control horizon N_u, which is defined as the time in which the actuating variable no longer changes (receding horizon). In this way the computation effort is greatly reduced. Only the first set of control actions is applied to the system, because in the next sampling instant a new prediction and optimization process is carried out. Based on the previous assumption, it is possible to notice that MPC tries to work in the same way as a human being. The control algorithm predicts the system behavior up to the time in which the control actions are effective. The control criterion depends on the cost function, which makes it possible to optimize the reference tracking or the control effort through adjustable weighting factors.

The computed estimations and predictions are based on a discrete-time system model. Thus, an accurate discretization method is needed to transform the continuous-time system model to a discrete-time model. The most simple expression to describe a continuous system through a linear discrete-time model is written in (2.1):

$$\mathbf{x}(k+1) = \mathbf{A_d}\mathbf{x}(k) + \mathbf{B_d}\mathbf{u}(k), \tag{2.1}$$

$$\mathbf{y}(k) = \mathbf{C_d}\mathbf{x}(k), \tag{2.2}$$

where, the system variables are limited by the following output constrains: $\mathbf{x} \in X \subset \mathbb{R}^n$, $\mathbf{y} \in Y \subset \mathbb{R}^q$ and the inputs are limited by the following input constrains: $\mathbf{u} \in U \subset \mathbb{R}^p$. A quadratic cost function G is defined in (2.3):

$$\begin{aligned} G(N_p) = {} & \mathbf{\Delta y}(k+N_p)^T \mathbf{P} \mathbf{\Delta y}(k+N_p) + \\ & \sum_{j=1}^{N_p} (\mathbf{\Delta y}(k+j-1)^T \mathbf{Q} \mathbf{\Delta y}(k+j-1) + \mathbf{u}(k+j-1)^T \mathbf{R} \mathbf{u}(k+j-1)), \end{aligned} \tag{2.3}$$

where, $\mathbf{\Delta y} = \mathbf{y}^* - \mathbf{y}$ is the tracking error vector, the matrix $\mathbf{Q}$ is weighting the output, $\mathbf{R}$ corresponds to the matrix, which penalizes the control actions and $\mathbf{P}$ is weighting the final values of the system output. Then, the optimization problem is

$$U_{opt} = \arg \min_{\mathbf{u} \in U} G(N_p). \tag{2.4}$$

Thus, it is necessary to compute (2.4), considering the following constrains:

$$\begin{aligned}
\mathbf{x}(k+j) &\in X,\ \ j = 1...N_p \\
\mathbf{u}(k+j) &\in U,\ \ j = 0...N_p \\
\mathbf{x}(k+j+1) &= \mathbf{A_d}\mathbf{x}(k+j) + \mathbf{B_d}\mathbf{u}(k+j),\ \ k >> 0 \\
\mathbf{u}(k+j) &= \mathbf{K}\mathbf{x}(k+j),\ \ N_u < k < N_p \\
\mathbf{Q} &= \mathbf{Q}^T \geq 0,\ \ \text{i.e., positive semidefinite} \\
\mathbf{R} &= \mathbf{R}^T > 0,\ \ \text{i.e., positive definite} \\
\mathbf{P} &\geq 0.
\end{aligned} \tag{2.5}$$

The optimization presented in (2.5) can be transformed to a Quadratic Programming Problem (QPP). Due to the constrains, the optimization cannot be calculated online in only one step. Hence, it is necessary to solve it in a multistep program [54]. Despite the fact that MPC is a very powerful control strategy. The minimization of the quadratic programming equation takes to much execution time if the algorithm is implemented to work online. This fact makes the application of MPC with a long prediction horizons difficult in the control of electrical drives [16]. However, it is possible to reduce the amount of calculation time in the solution of the quadratic programming problem of MPC noticing that the solution of the optimization problem is piecewise affine (PWA) over the state space [54]. Almost all the control effort can be computed in an offline way by using the state vector as a parameter of optimization, instead of a direct state variable.

Applying this explicit control law to the system, it is necessary to find the sub-space where the actual system state is located. This controller has the same properties as MPC and computational effort is greatly reduced. The mathematical background to be able to understand this solution is quite hard, but the Zurich University (ETH) has developed a MATLAB toolbox to deal with this problem, getting the explicit solutions of MPC [55]. Some application for electrical drives are presented in [56].

Unfortunately, the use of a precalculated piecewise affine control law, instead of solving the quadratic programming problem online, it is still not able to make MPC feasible for online drive control. This is caused by the fact that the different PWA regions of the explicit control law are not sorted at all; therefore it is necessary to perform an exhaustive search over all regions in order to find the active one. Another solution is to transform the problem into a binary search tree [30]. This allows a faster evaluation of the PWA control law, which makes the application of MPC to the control of electrical drives feasible.

2.4.2 Finite Control Set Model Predictive Control

2.4.2.1 Large Prediction Horizon

The first element of the method is to count with a model of the system to control. This model is in general non-linear due it includes the switching nature of the converter. A well-known discrete-time representation is

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)), \tag{2.6}$$

$$\mathbf{y}(k) = h(\mathbf{x}(k), \mathbf{u}(k)), \tag{2.7}$$

where, $\mathbf{x}(k)$ represents the states vector at time k and $\mathbf{u}(k)$ the control-action vector or directly the voltage vector to apply by the converter ($\mathbf{u}(k) \equiv \mathbf{S}(k)$). The system (2.6)-(2.7) represents a bilinear time-invariant discrete-time dynamic model [57]. Then, a general formulation of a cost or quality function that represents the criterion to evaluate the behavior of the system is

$$G(N_p) = F(\mathbf{\Delta y}(k+N_p) + \sum_{j=k}^{k+N_p-1} L(\mathbf{\Delta y}(j), \mathbf{u}(j)), \tag{2.8}$$

with U is the future and finite control-actions sequence defined as

$$U = \{\mathbf{u}(k), \mathbf{u}(k+1), ..., \mathbf{u}(k+N_p-1)\}, \tag{2.9}$$

$L(\cdot,\cdot)$ is a convex function of present and future states, control actions and reference values; $F(\cdot)$ explicitly considers a cost assigned to the final state of the system after N_p sampling intervals. The optimization problem is to find the value of U such that the optimum sequence of control actions is determined

$$U_{opt} = \arg\min_{\mathbf{u}\in U} G(N_p). \tag{2.10}$$

Thus, the optimum control action to apply is determined as the first element of the optimum sequence

$$\mathbf{u}(k) = \mathbf{u}(k)_{opt}. \tag{2.11}$$

2.4.2.2 Short Prediction Horizon

The above definitions are generalized for a large prediction horizon ($N_p \geq 2$). However, when the system has large number of voltage vectors or the control objectives are more than three, the computational burden easily grows. Most applications of FCS-MPC in power electronics and drives use only horizon-one cost functions [31]. Interestingly, in some situations, the use of horizon one ($N_p = 1$) also gives the optimal solution to a formulation with a larger horizon [37]. In the next chapters, good performance will be obtained with short prediction horizons. Finally, when a horizon-one is selected and quadratic cost function is considered, the quality function that represents the criterion to evaluate the behavior of the system is

$$G = \mathbf{\Delta y}(k+1)^T \mathbf{P} \mathbf{\Delta y}(k+1), \tag{2.12}$$

where, $\mathbf{P} = diag\{k_1, k_2, ..., k_q\}$ is the well-know weighting factor matrix associated to each output (objective).

2.5 FCS-MPC Formulation for Drive Applications

The increasing number of drive applications, in which fast dynamic response and algorithm simplicity are required, has demanded the development of new control strategies capable to improve their performance [34]. Furthermore, traditional controllers do not have any information related to the plant. This is only required during their offline design.

Due to the fast development of microprocessors, the idea of having only a single centralized controller, without a cascade control structure, was considered in order to improve the dynamic torque response. The concept of FCS-MPC is based on the calculation of the future behavior of the system, in order to use this information to compute optimal values for the actuating variables. Finally, the FCS-MPC formulation is based in three stages: modeling, cost function selection and optimization algorithm.

2.5.1 System Model

Modeling is the fundamental step of FCS-MPC scheme, with an accurate model is possible to determine the actuation effect over the state variable evolution. Lets remember that state variables evolution are modeled in a continuous time (obtained with fundamental circuital laws), while in a practical implementation it is commonly calculated in a digital platform (i.e., in discrete form). Then, an accurate discrete-time model is needed. The future behavior of the system is predicted using a discrete-time representation of the continuous-time model of the system, where the discrete-time model is obtained using an accurate discretization method. Usually, Forward or Backward Euler method is used in power electronics but it is not the best choice [58]. More details about this issue is explained with details in the next chapter.

2.5.2 Cost Function Selection

The most common terms in a cost function are the ones that represent a variable following a reference. Some examples are current control, torque control, power control, etc (Table 1.1). These terms can be expressed in a general way as the error between the predicted variable and its reference:

$$G_{quadratic} = \mathbf{\Delta y}(k+1)^T \mathbf{P} \mathbf{\Delta y}(k+1), \tag{2.13}$$

$$G_{absolute} = \sum_{i=1}^{q} k_i |y_i(k+1) - y_i^*(k+1)|, \tag{2.14}$$

where $G_{quadratic}$ and $G_{absolute}$ is a quadratic and absolute cost function, respectively. Squared error presents a better reference following when additional terms are included in the cost, for this reason a quadratic cost function is used in the next sections.

In a real-time implementation the time required to compute the control law algorithm may take a significant portion of the sample period, resulting in one sampling time delay. This phenomenon is well understood and may be compensated. In the FCS-MPC scheme the effect of the one delay has a large impact on the prediction, especially when an horizon-one algorithm is considered, and therefore a delay compensation scheme must be implemented. Here, a model-based prediction is used to compensate the calculation delay, the first step is an extrapolation used to estimate the output in $k+1$ time, which is used as an initial condition for the predictions in $k+2$ sampling time [32, 34, 59]. Thus, the optimal actuation is selected as

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}(k+1) \in U} \mathbf{\Delta y}(k+2)^T \mathbf{P} \mathbf{\Delta y}(k+2). \tag{2.15}$$

It should be noted that (2.13) can be expressed as a linear combination of independent objective functions,

$$\begin{aligned} G &= \mathbf{\Delta y}(k+2)^T \mathbf{P} \mathbf{\Delta y}(k+2), \\ &= \sum_{i=1}^{q} k_i g_i, \\ &= k_1 g_1 + k_2 g_2 + \ldots + k_q g_q, \end{aligned} \tag{2.16}$$

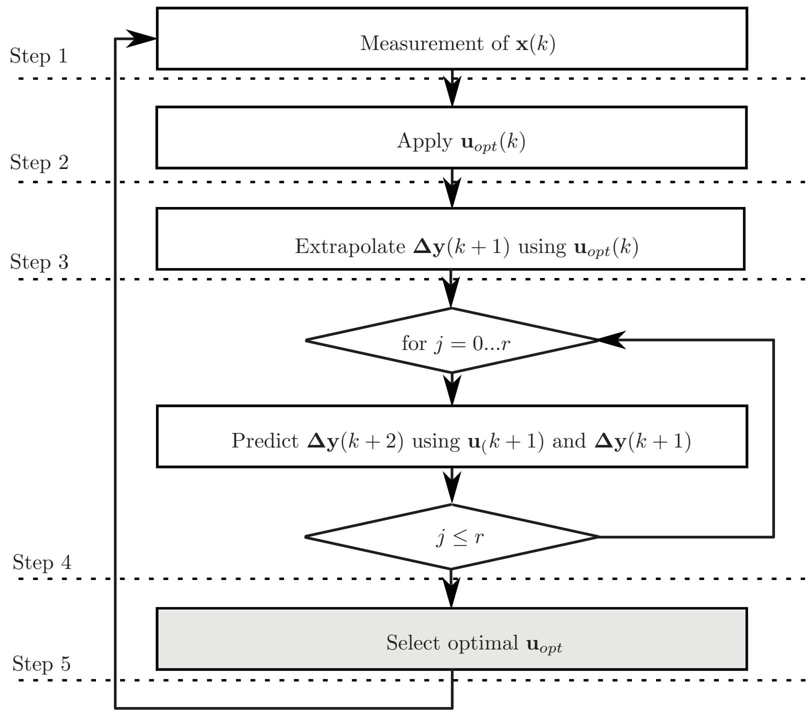


Figure 2.5: General flux diagram of FCS-MPC algorithm.

where, k_i with $i = 1, \ldots, q$ are the weighting factor associated to each control objective $g_i = (\Delta y_i(k+2))^2$. Finally, input and output constraints can be included, such as switching frequency limitation, variable limitation (saturations), common-mode voltage reduction, etc. Weighting factors must ensure that all objectives are achieved satisfactorily. The difference between the objectives nature, i.e., dynamic and variation range, makes the weighting factor determination a non-trivial task [46].

2.5.3 Optimization Algorithm

The minimization of (2.13) is performed by an exhaustive search for all feasible converter actuation. The proposed control strategy can be described in the following sequence:

- Step 1 *Measurement*: Sampling to get measurable state variables $\mathbf{x}(k)$.
- Step 2 *Apply*: Set the optimal actuation $\mathbf{u}_{opt}(k)$ found in the previous loop iteration.
- Step 3 *Extrapolate*: Extrapolate the discrete-time model using $\mathbf{u}_{opt}(k)$ to estimate $\mathbf{\Delta y}(k+1)$.
- Step 4 *Predict*: Predict the control variables for every possible actuation vector $\mathbf{u}_{(}(k+1)$, using $\mathbf{\Delta y}(k+1)$ as an initial condition for $\mathbf{\Delta y}(k+2)$.
- Step 5 *Optimize*: Select optimal $\mathbf{u}_{opt}$. Return to Step 1.

The conventional FCS-MPC scheme for a converter with r feasible switching states or voltage vectors is illustrated in Fig. 2.5. In drive applications there exist some variables which its measurement is a hard or unpractical, e.g., measurement of fluxes in an induction machine. For this reason, an *estimation* step is needed in the algorithm of Fig. 2.5.

2.6 Conclusions

Some of the most important characteristics of MPC for Power Electronics and drives have been reviewed in this chapter. The increasing attention given to FCS-MPC in this field is remarkable, as reflected in its implementation in a wide range of power topologies and applications [31]. These advances have been made possible in great part by the availability of modern digital control platforms, whose ever-increasing computing power is making the research of more elaborate FCS-MPC techniques possible.

Several works reported in the recent literature have demonstrated that predictive schemes are an alternative to the classical control solutions, being generally superior in terms of transient performance and flexibility. A number of recent studies have aimed to mitigate some of the drawbacks of the FCS-MPC schemes, such as variable switching frequency and the need for tuning weighting factors, achieving promising results. Also, the flexibility of the FCS-MPC technology has motivated a great number of novel and interesting proposals for addressing practical problems in the field of Power Electronics.

Chapter 3

INDUCTION MACHINE DRIVE MODELING

3.1 Introduction

SYSTEM modeling and variables identification are two important task in FCS-MPC schemes, for this reason the hybrid system model must be accurately obtained. The studied system is based on a conventional two-level voltage source drive, where the load is an induction machine (IM) fed by a classical two-level inverter (2L-VSI). The selection of a conventional drive suggests that if the proposed algorithms work for a conventional drive, there is no theoretical impediment to make it operate on more complex systems such as Multilevel Drives or Matrix Converters [47].

3.2 Converter Model

The topology used to feed the IM is a classical Two-Level Inverter (2L-VSI) shown in Fig. 3.1(a). This topology has eight possible switching states (finite states), which produce seven different voltage vectors illustrated in Fig. 3.1(b). Six of them ($\mathbf{v_1}$ to $\mathbf{v_6}$) are active vectors and the other two ($\mathbf{v_0}$ and $\mathbf{v_7}$) are called zero vectors. The possible switching states of a 2L-VSI are summarized in Table 3.1.

The voltage and current variables of the three-phase system illustrated in Fig. 3.1(a) can be represented by a three-axis coordinate system,

$$\mathbf{v_s}_{abc} = \begin{bmatrix} v_{sa} & v_{sb} & v_{sc} \end{bmatrix}^T, \tag{3.1}$$

$$\mathbf{i_s}_{abc} = \begin{bmatrix} i_{sa} & i_{sb} & i_{sc} \end{bmatrix}^T. \tag{3.2}$$

Furthermore, the relation between the output voltage applied to the machine with respect to the dc-link voltage v_{dc} and switching functions S_a, S_b and S_c is,

$$\mathbf{v_s}_{abc} = \begin{bmatrix} S_a & S_b & S_c \end{bmatrix}^T v_{dc}. \tag{3.3}$$

Now, considering the system without a neutral wire connection, it can be represented by

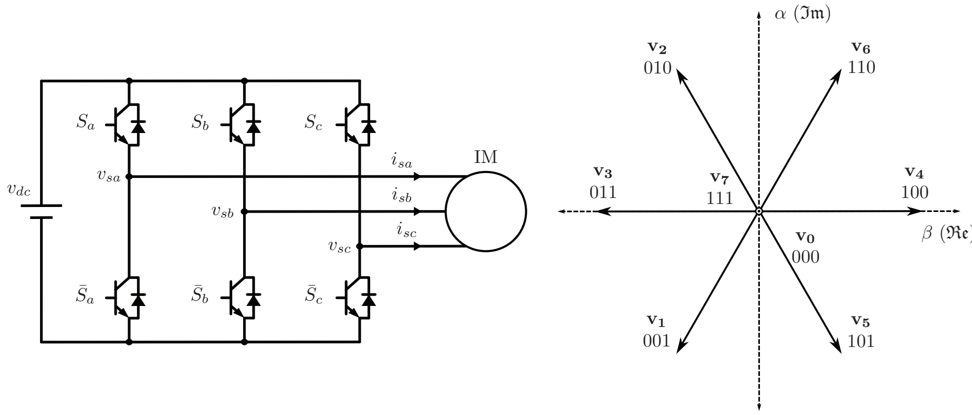


Figure 3.1: Two-level inverter and its voltage vectors.

Table 3.1: Possible switching states of a 2L-VSI

Voltage Vector	Switching State S_a	S_b	S_c	Voltage $\alpha\beta$-frame $v_{s\alpha}$	$v_{s\beta}$
$\mathbf{v_0}$	0	0	0	0	0
$\mathbf{v_1}$	1	0	0	$\frac{2}{3}v_{dc}$	0
$\mathbf{v_2}$	1	1	0	$\frac{1}{3}v_{dc}$	$\frac{1}{\sqrt{3}}v_{dc}$
$\mathbf{v_3}$	0	1	0	$-\frac{1}{3}v_{dc}$	$\frac{1}{\sqrt{3}}v_{dc}$
$\mathbf{v_4}$	0	1	1	$-\frac{2}{3}v_{dc}$	0
$\mathbf{v_5}$	0	0	1	$-\frac{1}{3}v_{dc}$	$-\frac{1}{\sqrt{3}}v_{dc}$
$\mathbf{v_6}$	1	0	1	$\frac{1}{3}v_{dc}$	$-\frac{1}{\sqrt{3}}v_{dc}$
$\mathbf{v_7}$	1	1	1	0	0

only two variables,

$$\mathbf{v}_{\alpha\beta} = \begin{bmatrix} v_{s\alpha} & v_{s\beta} \end{bmatrix}^T = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\mathbf{v_s}_{abc}, \tag{3.4}$$

where the axis $\alpha\beta$ are the well-know stationary frame. Another well-known representation is achieved with respect to an arbitrary system variable oriented with θ_f,

$$\mathbf{v}_{dq} = \begin{bmatrix} v_{sd} & v_{sq} \end{bmatrix}^T = \begin{bmatrix} cos(\theta_f) & sin(\theta_f) \\ -sin(\theta_f) & cos(\theta_f) \end{bmatrix}\mathbf{v_s}_{\alpha\beta}, \tag{3.5}$$

where the axis dq are the well-know rotating frame at $\omega_f = \dot{\theta}_f$.

3.3 Induction Machine Model

Induction machine can be represented in any arbitrary reference frame, rotating with an angular frequency ω_f, as it is shown in the followings equations,

$$\mathbf{v_s} = R_s\mathbf{i_s} + \frac{d\mathbf{\Psi_s}}{dt} - \omega_f\mathbf{J}\mathbf{\Psi_s}, \tag{3.6}$$

$$0 = R_r\mathbf{i_r} + \frac{d\mathbf{\Psi_r}}{dt} + (\omega_f - \omega)\,\mathbf{J}\mathbf{\Psi_r}, \tag{3.7}$$

$$\mathbf{\Psi_s} = L_s\mathbf{i_s} + L_m\mathbf{i_r}, \tag{3.8}$$

$$\mathbf{\Psi_r} = L_m\mathbf{i_s} + L_r\mathbf{i_r}, \tag{3.9}$$

$$T = \frac{3}{2}p\left(\mathbf{\Psi_s} \times \mathbf{i_s}\right), \tag{3.10}$$

$$J\frac{d\omega}{dt} = T - T_l, \tag{3.11}$$

with $\mathbf{v_s}$ the stator voltage vector, $\mathbf{i_s}$ the stator current vector, $\mathbf{i_r}$ the rotor current vector, $\mathbf{\Psi_s}$ the stator flux vector, $\mathbf{\Psi_r}$ the rotor flux vector, T the electrical torque, T_l the load torque and $\mathbf{J}$ an ortogonal matrix [60] defined by

$$\mathbf{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \tag{3.12}$$

The set $\{R_s,\ R_r,\ L_s,\ L_r,\ L_m,\ J,\ p\}$ are the system parameters. The variable ω denotes the rotor angular speed. This model is obtained from the simplified single fase equivalent model of the induction machine [61]. Finally, from the control point of view, the induction machine is summarized by

- $\mathbf{v_s}$: input or actuation vector
- $\mathbf{i_s}$: measurable variable
- $\mathbf{\Psi_r}$, $\mathbf{\Psi_s}$: non-measurable or estimated variables
- $\{R_s,\ R_r,\ L_s,\ L_r,\ L_m,\ J,\ p\}$: constant system parameters
- ω: time-varying parameter
- ω_f: time-varying parameter, synchronism variable

3.3.1 Stationary-Frame

If the arbitrary reference frame is a stationary frame commonly named $\alpha\beta$-frame the arbitrary angular frequency is $\theta_f = 0$, i.e., $\omega_f = 0$. Then, the IM model equations are

$$\mathbf{v_s}_{\alpha\beta} = R_s\mathbf{i_s}_{\alpha\beta} + \frac{d\mathbf{\Psi_s}_{\alpha\beta}}{dt}, \tag{3.13}$$

$$0 = R_r\mathbf{i_r}_{\alpha\beta} + \frac{d\mathbf{\Psi_r}_{\alpha\beta}}{dt} - \omega\mathbf{J}\mathbf{\Psi_r}_{\alpha\beta}, \tag{3.14}$$

$$\mathbf{\Psi_s}_{\alpha\beta} = L_s\mathbf{i_s}_{\alpha\beta} + L_m\mathbf{i_r}_{\alpha\beta}, \tag{3.15}$$

$$\mathbf{\Psi_r}_{\alpha\beta} = L_m\mathbf{i_s}_{\alpha\beta} + L_r\mathbf{i_r}_{\alpha\beta}. \tag{3.16}$$

The dynamic of the stator current $\mathbf{i_s}_{\alpha\beta}$ is obtained using the equivalent equation of the stator dynamics of a squirrel-cage induction machine [6]:

$$\tau_\sigma \frac{d\mathbf{i_s}_{\alpha\beta}}{dt} + \mathbf{i_s}_{\alpha\beta} = \frac{k_r}{R_\sigma}\left(\frac{1}{\tau_r}\mathbf{I} - \omega\mathbf{J}\right)\mathbf{\Psi_r}_{\alpha\beta} + \frac{1}{R_\sigma}\mathbf{v_s}_{\alpha\beta}, \tag{3.17}$$

where $\mathbf{I}$ is the identity matrix in $\mathbb{R}^{4\times4}$, $R_\sigma = R_s + k_r^2 R_r$ corresponds to the equivalent resistance, $\tau_\sigma = \frac{L_\sigma}{R_\sigma}$ is the transient time stator constant, $\tau_r = \frac{L_r}{R_r}$ is the rotor time constant, $L_\sigma = \sigma L_s$ is the leakage inductance of the machine, $k_r = \frac{L_m}{L_r}$ is the rotor coupling factor and $\sigma = 1 - \frac{L_m^2}{L_s L_r}$ is the total leakage factor.

3.3.2 Rotatory-Frame

Now, if the arbitrary reference frame is a synchronous frame with the rotor flux angle with respect to the stator winding θ_s, commonly called dq-frame. Then, the arbitrary angular frequency is $\omega_f = \omega_s$, with $\omega_s = \dot{\theta}_s$.

$$\mathbf{v_s}_{dq} = R_s\mathbf{i_s}_{dq} + \frac{d\mathbf{\Psi_s}_{dq}}{dt} - \omega_s\mathbf{J}\mathbf{\Psi_s}_{dq}, \tag{3.18}$$

$$0 = R_r\mathbf{i_r}_{dq} + \frac{d\mathbf{\Psi_r}_{dq}}{dt} + (\omega_s - \omega)\,\mathbf{J}\mathbf{\Psi_r}_{dq}, \tag{3.19}$$

$$\mathbf{\Psi_s}_{dq} = L_s\mathbf{i_s}_{dq} + L_m\mathbf{i_r}_{dq}, \tag{3.20}$$

$$\mathbf{\Psi_r}_{dq} = L_m\mathbf{i_s}_{dq} + L_r\mathbf{i_r}_{dq}. \tag{3.21}$$

Then, by using a rotatory-frame, the dynamic of the stator current $\mathbf{i_s}_{dq}$ is obtained using the equivalent equation of the stator dynamics of a squirrel-cage induction machine [6]

$$\tau_\sigma \frac{d\mathbf{i_s}_{dq}}{dt} = -\left(\mathbf{I} + \omega_s\tau_\sigma\mathbf{J}\right)\mathbf{i_s}_{dq} + \frac{k_r}{R_\sigma}\left(\frac{1}{\tau_r}\mathbf{I} - \omega\mathbf{J}\right)\mathbf{\Psi_r}_{dq} + \frac{1}{R_\sigma}\mathbf{v_s}_{dq}. \tag{3.22}$$

3.3.3 System Variables

The model (3.6)-(3.11) can be separated on two subsystems: electromagnetic system given by (3.6)-(3.9) and mechanical system (3.10)-(3.11). In conventional FCS-MPC schemes for drive applications the cascade control structure of FOC is retained, where the mechanical system is controlled by conventional linear PI-controllers, while internal current loop is controlled using an optimization stage. However, there are some schemes where the induction machine is fully controlled by using FCS-MPC [25, 31].

In this work the speed dynamic is controlled using a PI-controller, where the speed control-loop bandwidth is lower than the inner loop. Remember that speed is a mechanical variable, it is slower than electrical variables. Furthermore, FCS-MPC has a very high bandwidth allowing a high decoupling degree between inner and external loop [34]. Finally, if the mechanical system is controlled by a PI-controller, the study of the electromagnetic sub-system is only needed.

From (3.6)-(3.9) is possible to note that there are two state vectors only: stator $\mathbf{\Psi_s}$ and rotor $\mathbf{\Psi_r}$ fluxes, respectively. However, for both variables an estimation stage is needed. Commonly, one of them is replaced by stator current vector due to stator currents are directly measurable. Commonly rotor flux $\mathbf{\Psi_r}$ and stator current vector $\mathbf{i_s}$ are used as state variables. Then, the system variable identification is developed using the above variables in stationary frame:

$$\mathbf{u} = \mathbf{v}_{\mathbf{s}\alpha\beta} = [v_{s\alpha}\ v_{s\beta}]^T, \tag{3.23}$$
$$\mathbf{x} = [\mathbf{i}_{\mathbf{s}\alpha\beta}^T\ \mathbf{\Psi}_{\mathbf{r}\alpha\beta}^T]^T = [i_{s\alpha}\ i_{s\beta}\ \psi_{r\alpha}\ \psi_{r\beta}]^T, \tag{3.24}$$
$$\mathbf{y} = [\mathbf{i}_{\mathbf{s}\alpha\beta}^T\ \mathbf{\Psi}_{\mathbf{s}\alpha\beta}^T]^T = [i_{s\alpha}\ i_{s\beta}\ \psi_{s\alpha}\ \psi_{s\beta}]^T. \tag{3.25}$$

3.4 Continuous-Time Model

Considering a stator reference frame, the continuous-time electrical model of the induction machine is given by a set of four linearly independent equations in terms of their $\alpha\beta$-components, that correspond to a linear time-varying (LTV) system:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t), \tag{3.26}$$
$$\mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t), \tag{3.27}$$

where $\mathbf{x} = [i_{s\alpha}\ i_{s\beta}\ \psi_{r\alpha}\ \psi_{r\beta}]^T$ is the internal state vector, $\mathbf{u} = [v_{s\alpha}\ v_{s\beta}]^T$ is the input vector, and $\mathbf{y} = [i_{s\alpha}\ i_{s\beta}\ \psi_{s\alpha}\ \psi_{s\beta}]^T$ is the input vector. Equation (3.26) describes the system dynamics, whereas (3.27) describes how the available measurements are obtained from the internal variables. The matrices $\mathbf{A}(t)$, $\mathbf{B}(t)$, $\mathbf{C}(t)$ and $\mathbf{D}(t)$ are state-space matrices, given by

$$\mathbf{A}(t) = \mathbf{A}(\omega(t)) = \begin{bmatrix} -\frac{1}{\tau_\sigma} & 0 & \frac{k_r}{R_\sigma \tau_\sigma \tau_r} & \frac{k_r}{R_\sigma \tau_\sigma}\omega(t) \\ 0 & -\frac{1}{\tau_\sigma} & -\frac{k_r}{R_\sigma \tau_\sigma}\omega(t) & \frac{k_r}{R_\sigma \tau_\sigma \tau_r} \\ \frac{L_m}{\tau_r} & 0 & -\frac{1}{\tau_r} & -\omega(t) \\ 0 & \frac{L_m}{\tau_r} & \omega(t) & -\frac{1}{\tau_r} \end{bmatrix}, \tag{3.28}$$

$$\mathbf{B}(t) = \begin{bmatrix} \frac{1}{R_\sigma \tau_\sigma} & 0 \\ 0 & \frac{1}{R_\sigma \tau_\sigma} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{3.29}$$

$$\mathbf{C}(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ L_\sigma & 0 & k_r & 0 \\ 0 & L_\sigma & 0 & k_r \end{bmatrix}, \tag{3.30}$$

$$\mathbf{D}(t) = \mathbf{0}. \tag{3.31}$$

Parameter variations can also have a negative impact on model-based control strategies. Nevertheless, in this research the parameters are assumed to be known, balanced and constant [28,46]. Finally, from (3.26) and (3.10) by selecting a suitable discretization method is possible to affirm that both stator flux $\mathbf{\Psi}_{\mathbf{s}\alpha\beta}$ and electromagnetic torque T can be modified by the control actuation or stator voltage vector $\mathbf{v}_{\mathbf{s}\alpha\beta}$ such as the conventional DTC scheme.

3.5 Discrete-Time Model

Note that in (3.29)-(3.30) matrices $\mathbf{B}(t) = \mathbf{B}$, and $\mathbf{C}(t) = \mathbf{C}$ are constant. However, the matrix $\mathbf{A}(t)$ in (3.28) depends on the instantaneous value of the mechanical speed $\omega(t)$, making $\mathbf{A}(t) = \mathbf{A}(\omega(t))$ a linear time-varying matrix. Time dependence means that an exact sampled-data model of this system is not straightforward to obtain due to the variations in $\omega(t)$ [16]. To obtain a discrete-time description from the state-space model (3.26)-(3.27) a Zero-Order Hold (ZOH) input is assumed, i.e.,

$$\mathbf{u}(t) = \mathbf{u}^k, \forall\ t\ \in\ [kT_s, (k+1)T_s], \tag{3.32}$$

where $\mathbf{u}^k = \mathbf{u}(k)$ is the input of the system, T_s is the sampling period. Then, the set of differential equations can be solved, leading to the following discrete-time model:

$$\mathbf{x}^{k+1} = \mathbf{A_d}^k \mathbf{x}^k + \mathbf{B_d}^k \mathbf{u}^k, \tag{3.33}$$

$$\mathbf{y}^k = \mathbf{C_d}^k \mathbf{x}^k + \mathbf{D_d}^k \mathbf{u}^k, \tag{3.34}$$

where the matrices are:

$$\mathbf{A_d}^k = e^{\int_{kT_s}^{(k+1)T_s} \mathbf{A}(\tau)d\tau}, \tag{3.35}$$

$$\mathbf{B_d}^k = \int_{kT_s}^{(k+1)T_s} e^{\int_{\eta}^{(k+1)T_s} \mathbf{A}(\tau)d\tau} \mathbf{B}(\eta)d\eta, \tag{3.36}$$

$$\mathbf{C_d}^k = \mathbf{C}, \tag{3.37}$$

$$\mathbf{D_d}^k = \mathbf{D}. \tag{3.38}$$

The model given by (3.33)-(3.34) is an exact discrete-time model, in the sense that the continuous-time state and output (3.26)-(3.27) are exactly recovered at the k-th sampling instant, where $\mathbf{y}^k = \mathbf{y}(kT_s)$ and $\mathbf{x}^k = \mathbf{x}(kT_s)$. However, for nonlinear or time-varying models (as in the case of the induction machine) exact sampled-data models may be hard or impossible to obtain explicitly since this requires solving (3.35)-(3.36) online.

A common approximation to obtain a discrete-time model for the induction machine is to assume that the changes in the mechanical speed are slow with respect to the sampling time and, therefore, it can be considered to be constant within each sampling period [60], i.e.,

$$\omega(t) = \omega^k, \forall\ t\ \in\ [kT_s, (k+1)T_s]. \tag{3.39}$$

Then, matrices (3.35)-(3.36) can be simplified to obtain

$$\mathbf{A_d}^k = e^{\mathbf{A}T_s}, \tag{3.40}$$

$$\mathbf{B_d}^k = \int_0^{T_s} e^{\mathbf{A}\eta}\mathbf{B}d\eta = \mathbf{A}^{-1}(e^{\mathbf{A}T_s} - I)\mathbf{B}, \tag{3.41}$$

$$\mathbf{C_d}^k = \mathbf{C}, \tag{3.42}$$

$$\mathbf{D_d}^k = \mathbf{D}, \tag{3.43}$$

with $\mathbf{A} = \mathbf{A}(\omega^k)$ defined in (3.28) and ω^k is the measured value of $\omega(t)$ at sampling time kT_s. Note that, the inverse of $\mathbf{A}$ is nonsingular due to $L_m k_r - R_\sigma \tau_r = -\frac{L_r R_s}{R_r}$

$$det(\mathbf{A}) = \frac{\left(\omega^2\tau_r^2 + 1\right)\left(L_m k_r - R_\sigma \tau_r\right)^2}{\left(R_\sigma \tau_r^2 \tau_\sigma\right)^2}. \tag{3.44}$$

Given the time-varying nature of the model (as explained above), the instantaneous values of the matrices $\mathbf{A_d}^k$ and $\mathbf{B_d}^k$ have to be updated at every sampling time. In order to obtain these matrices, the matrix exponential $e^{\mathbf{A}T_s}$ has to be computed at every sampling interval using the measured value of ω^k. This calculation is a highly time-consuming process and thus approximations such as Euler or Tustin are commonly used [16]. However, Euler discretization may lead to poor accuracy for real-time control [58, 62, 63].

In the following subsections several discretization methods are introduced, specifically, Euler, Taylor and Cayley-Hamilton are analyzed in depth.

3.5.1 Forward-Euler (Euler)

Probably the most common and simple method to obtain a discrete-time representation for a continuous-time system is the Euler approximation of time derivatives. The obtained model corresponds to a first order Taylor expansion,

$$x_{k+1} = x_k + T_s \left.\dot{x}\right|_k . \tag{3.45}$$

Using Euler in (3.26)-(3.27), a discretized state-space model is derived

$$\mathbf{A_d}^k{}_{Eu} = I + T_s\mathbf{A}, \tag{3.46}$$
$$\mathbf{B_d}^k{}_{Eu} = T_s\mathbf{B}, \tag{3.47}$$
$$\mathbf{C_d}^k{}_{Eu} = \mathbf{C}, \tag{3.48}$$
$$\mathbf{D_d}^k{}_{Eu} = \mathbf{D}, \tag{3.49}$$

where Eu stands for Euler approximation and I is the identity matrix in $\mathbb{R}^{4\times 4}$. An issue related to the use of Forward-Euler approximation is the relative degree of the resultant discrete-time model, which may lead to poor horizon-one prediction as discussed in [58]. Finally, by using a Euler approximation the resultant discrete-time model is presented analytically in (3.50)-(3.53).

$$\mathbf{A_d}^k{}_{Eu} = \begin{bmatrix} 1\text{-}\frac{T_s}{\tau_\sigma} & 0 & \frac{T_s k_r}{R_\sigma \tau_\sigma \tau_r} & \frac{T_s k_r}{R_\sigma \tau_\sigma}\omega^k \\ 0 & 1\text{-}\frac{T_s}{\tau_\sigma} & \text{-}\frac{T_s k_r}{R_\sigma \tau_\sigma}\omega^k & \frac{T_s k_r}{R_\sigma \tau_\sigma \tau_r} \\ \frac{T_s L_m}{\tau_r} & 0 & 1\text{-}\frac{T_s}{\tau_r} & \text{-}T_s\omega^k \\ 0 & \frac{T_s L_m}{\tau_r} & T_s\omega^k & 1\text{-}\frac{T_s}{\tau_r} \end{bmatrix}, \tag{3.50}$$

$$\mathbf{B_d}^k{}_{Eu} = \begin{bmatrix} \frac{T_s}{R_\sigma \tau_\sigma} & 0 \\ 0 & \frac{T_s}{R_\sigma \tau_\sigma} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{3.51}$$

$$\mathbf{C_d}^k{}_{Eu} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ L_\sigma & 0 & k_r & 0 \\ 0 & L_\sigma & 0 & k_r \end{bmatrix}, \tag{3.52}$$

$$\mathbf{D_d}^k{}_{Eu} = \mathbf{0}. \tag{3.53}$$

3.5.2 Tustin

Other usual approximations of the time derivatives is the Tustin approximation or bilinear transform. In this case, the time derivatives are approximated using the transformation presented in (3.54).

$$s = \frac{2}{T_s}\frac{z-1}{z+1}. \tag{3.54}$$

Then, using (3.54) and (3.26)-(3.27) and after several mathematical arrangements the discretized state-space model is derived

$$\mathbf{A_d}^k{}_{Tu} = \left(\mathbf{I} - \frac{T_s}{2}\mathbf{A}\right)^{-1}\left(\mathbf{I} + \frac{T_s}{2}\mathbf{A}\right), \tag{3.55}$$

$$\mathbf{B_d}^k{}_{Tu} = T_s\left(\mathbf{I} - \frac{T_s}{2}\mathbf{A}\right)^{-1}\mathbf{B}, \tag{3.56}$$

$$\mathbf{C_d}^k{}_{Tu} = \mathbf{C}\left(\mathbf{I} - \frac{T_s}{2}\mathbf{A}\right)^{-1}, \tag{3.57}$$

$$\mathbf{D_d}^k{}_{Tu} = \frac{T_s}{2}\mathbf{C}\left(\mathbf{I} - \frac{T_s}{2}\mathbf{A}\right)^{-1}\mathbf{B} + \mathbf{D}. \tag{3.58}$$

The online calculation of the exact discrete-time system model is a highly time-consuming process, thus approximations such as Euler and Taylor are commonly used due to their state matrices can be calculated online easily for both methods. However, Tustin approximation has two main drawbacks, the first is the computational burden due to it involve matrix inversion and matrix multiplication, which increase the sampling time [63]. The second issue is that the matrix is non-zero $\mathbf{D_d}^k{}_{Tu} \neq \mathbf{0}$ introducing an important modification from the point of view of the continuous-time model, as the discrete-time model outputs are modified by the actuation.

3.5.3 Matrix Factorization

In [28], a predictive torque control strategy based on a sampled-data time-varying state-space induction machine model has been presented. In particular, the time-varying nature of the model is due to the rotor speed that appears in the state-space matrices of the machine electrical model. To obtain a sampled state-space model, in [28] the authors propose a factorization of the matrix exponential that does not hold for the continuous-time matrices associated with the machine model. Thus, the obtained discrete-time model is not exact. However, the use of the factorization proposed in [28] showed good results and a clear improvement in the proposed FCS-MPC strategy compared to the results obtained using the Euler approximation. Furthermore, similar results have been reported by other authors using this technique [64, 65].

In this section we first clarify that the use of the model proposed in [28] has to be understood as an approximate discretization and, secondly, we analyze its accuracy compared to the Euler discretization approach.

In order to obtain the time-varying matrices (3.40) and (3.41), in [28] the matrix $\mathbf{A}(t)$ is separated into a constant matrix, $\mathbf{A_c}$ (that does not depend on $\omega(t)$) and $\mathbf{A}_\omega(t)$ whose elements depend on the rotor speed $\omega(t)$, i.e.,

$$\mathbf{A}(t) = \mathbf{A_c} + \mathbf{A}_\omega(t), \tag{3.59}$$

where

$$\mathbf{A_c} = \begin{bmatrix} -\frac{1}{\tau_\sigma} & 0 & \frac{k_r}{R_\sigma \tau_\sigma \tau_r} & 0 \\ 0 & -\frac{1}{\tau_\sigma} & 0 & \frac{k_r}{R_\sigma \tau_\sigma \tau_r} \\ \frac{L_m}{\tau_r} & 0 & -\frac{1}{\tau_r} & 0 \\ 0 & \frac{L_m}{\tau_r} & 0 & -\frac{1}{\tau_r} \end{bmatrix}, \tag{3.60}$$

$$\mathbf{A}_\omega(t) = \begin{bmatrix} 0 & 0 & 0 & \frac{k_r}{R_\sigma \tau_\sigma}\omega(t) \\ 0 & 0 & -\frac{k_r}{R_\sigma \tau_\sigma}\omega(t) & 0 \\ 0 & 0 & 0 & -\omega(t) \\ 0 & 0 & \omega(t) & 0 \end{bmatrix}. \tag{3.61}$$

Then, assuming $\omega(t) \approx \omega^k$ when $kT_s \le t < (k+1)T_s$ and using (3.59), in [28] the authors propose that (3.40) could be expressed as

$$\mathbf{A_d}^k{}_{MF} = e^{\mathbf{A}T_s} = e^{(\mathbf{A_c}+\mathbf{A}_\omega)T_s} = e^{\mathbf{A_c}T_s} e^{\mathbf{A}_\omega T_s}, \tag{3.62}$$

where MF stands for Matrix Factorization. The matrix $e^{\mathbf{A_c}T_s}$ can be calculated offline since it is constant. On the other hand, for a sampled rotor speed $\omega(t) = \omega^k$, it is possible to obtain an exact representation of $e^{\mathbf{A}_\omega T_s}$ using the Cayley-Hamilton theorem [66]. Then, using (3.62), the discretized state-space model proposed in [28] is derived

$$\mathbf{A_d}^k{}_{MF} = e^{\mathbf{A_c}T_s} e^{\mathbf{A}_\omega T_s}, \tag{3.63}$$

$$\mathbf{B_d}^k{}_{MF} = \int_0^{T_s} e^{\mathbf{A_c}\eta} e^{\mathbf{A}_\omega \eta} \mathbf{B} d\eta. \tag{3.64}$$

$$\mathbf{C_d}^k{}_{MF} = \mathbf{C}, \tag{3.65}$$

$$\mathbf{D_d}^k{}_{MF} = \mathbf{D}, \tag{3.66}$$

Furthermore, in our particular case, the computation of $\mathbf{B_d}^k{}_{MF}$ in (3.64) can be simplified due to the fact that $e^{\mathbf{A}_\omega \eta}\mathbf{B} = \mathbf{B}$, and then

$$\mathbf{B_d}^k{}_{MF} = \mathbf{A_c}^{-1}(e^{\mathbf{A_c}T_s} - \mathbf{I})\mathbf{B}, \tag{3.67}$$

thus, it can also be computed offline.

However, the last identity in equation (3.62) is not valid in general and, thus, the discretized state-space model proposed in (3.63)-(3.64) is not exact as suggested in [28]. This is evident from the following lemma,

Lemma 1. Let $\mathbf{A},\ \mathbf{B} \in \mathbb{R}^{n\times n}$. Then,

$$e^{\Delta\mathbf{A}} e^{\Delta\mathbf{B}} = e^{\Delta(\mathbf{A}+\mathbf{B})} \tag{3.68}$$

for all $\Delta \in [0, \infty)$ if and only if $\mathbf{AB} = \mathbf{BA}$.

Proof. See [67], proposition 11.11.5. □

In our case, for the induction machine electrical model, $\mathbf{A_c}\mathbf{A}_\omega \neq \mathbf{A}_\omega\mathbf{A_c}$. In fact,

$$\mathbf{A_c}\mathbf{A}_\omega = \omega^k \begin{bmatrix} 0 & 0 & 0 & -\frac{k_r(\tau_r+\tau_\sigma)}{R_\sigma \tau_r \tau_\sigma^2} \\ 0 & 0 & \frac{k_r(\tau_r+\tau_\sigma)}{R_\sigma \tau_r \tau_\sigma^2} & 0 \\ 0 & 0 & 0 & \frac{L_m k_r + R_\sigma \tau_\sigma}{R_\sigma \tau_r \tau_\sigma} \\ 0 & 0 & -\frac{L_m k_r + R_\sigma \tau_\sigma}{R_\sigma \tau_r \tau_\sigma} & 0 \end{bmatrix}, \tag{3.69}$$

$$\mathbf{A}_\omega \mathbf{A_c} = \omega^k \begin{bmatrix} 0 & \frac{L_m k_r}{R_\sigma \tau_r \tau_\sigma} & 0 & -\frac{k_r}{R_\sigma \tau_r \tau_\sigma} \\ -\frac{L_m k_r}{R_\sigma \tau_r \tau_\sigma} & 0 & \frac{k_r}{R_\sigma \tau_r \tau_\sigma} & 0 \\ 0 & -\frac{L_m}{\tau_r} & 0 & \frac{1}{\tau_r} \\ \frac{L_m}{\tau_r} & 0 & -\frac{1}{\tau_r} & 0 \end{bmatrix}. \tag{3.70}$$

Then, by the simple comparison of (3.69) and (3.70), both expressions are different and, as a consequence, (3.62) does not hold true. Despite the fact that equation (3.62) does not in general hold, it can be used as a good approximation. The term $e^{\mathbf{A}_\omega T_s}$ must be updated in every sampling cycle. To compute this matrix, it is necessary to use the Cayley-Hamilton theorem [66], which demonstrates that it is possible to write an exponential function $e^{\mathbf{A}_\omega T_s}$ in terms of a polynomial $\mathbf{p}$ of degree n, where $n = dim(\mathbf{A})$:

$$\mathbf{p} = \alpha_0 \mathbf{I} + \alpha_1 \mathbf{A} + \cdots + \alpha_{n-1} \mathbf{A}^{n-1} = \sum_{i=0}^{n-1} \alpha_i \mathbf{A}^i. \tag{3.71}$$

In [66] it is demonstrated that the eigenvalues $\lambda_0 \cdots \lambda_{n-1}$ of $\mathbf{A}T_s$ are solutions of (3.71), i.e.,

$$e^{\lambda_i} = \alpha_0 \lambda_i^0 + \alpha_1 \lambda_i^1 + \cdots + \alpha_{i-1} \lambda_i^{n-1},\ i \in [0, \ldots, n-1]. \tag{3.72}$$

In our case, the polynomial $\mathbf{p}$ has grade $n = 4$ and its coefficients α_0, α_1, α_2 and α_3 depend on the eigenvalues of $\mathbf{A}_\omega T_s$. The eigenvalues of the matrix $\mathbf{A}_\omega T_s$ can be computed by using equation (3.73):

$$det(\lambda \mathbf{I} - \mathbf{A}_\omega T_s) = 0. \tag{3.73}$$

Resolving (3.73), the eigenvalues are determined by:

$$\lambda_0 = 0, \tag{3.74}$$
$$\lambda_1 = 0, \tag{3.75}$$
$$\lambda_2 = j\omega T_s, \tag{3.76}$$
$$\lambda_3 = -j\omega T_s. \tag{3.77}$$

If all eigenvalues have multiplicity $m = 1$ this yields n independent equations. For eigenvalues with multiplicity $m > 1$ also the first $m-1$ derivatives of (3.72) must be evaluated. Replacing the obtained eigenvalues in equation (3.72), it is possible to compute the values of α_i.

$$\alpha_0 = 1, \tag{3.78}$$
$$\alpha_1 = 1, \tag{3.79}$$
$$\alpha_2 = \frac{1 - \cos(\omega T_s)}{(\omega T_s)^2}, \tag{3.80}$$
$$\alpha_3 = \frac{\omega T_s - \sin(\omega T_s)}{(\omega T_s)^3}. \tag{3.81}$$

Finally, the resulting expression of the matrix $e^{\mathbf{A}_\omega T_s}$ is written in (3.82):

$$e^{\mathbf{A}_\omega T_s} = \begin{bmatrix} 1 & 0 & \frac{k_r}{\sigma L_s}(1-\cos(\omega T_s)) & \frac{k_r}{\sigma L_s}\sin(\omega T_s) \\ 0 & 1 & -\frac{k_r}{\sigma L_s}\sin(\omega T_s) & \frac{k_r}{\sigma L_s}(1-\cos(\omega T_s)) \\ 0 & 0 & \cos(\omega T_s) & -\sin(\omega T_s) \\ 0 & 0 & \sin(\omega T_s) & \cos(\omega T_s) \end{bmatrix}. \tag{3.82}$$

Remark 1. A similar analysis as presented in the previous section can be performed if ${\mathbf{A_d}}^k$ is approximated as

$${\mathbf{A_d}}^k{}_{MF} = e^{\mathbf{A}_\omega T_s} e^{\mathbf{A_c} T_s}, \tag{3.83}$$

instead of (3.63). In this case the associated input matrix ${\mathbf{B_d}}^k{}_{MF}$ is more difficult to obtain analytically compared to (3.67). On the other hand, the numerical analysis for this alternative approach shows results that are qualitatively similar to the results presented in Fig. 3.2a and Fig. 3.2b. However, the numerical value of $\mathcal{E}_{MF}^{{\mathbf{B_d}}^k}(T_s)$ in (3.100) is higher. □

3.5.4 Taylor

More accurate sampled-data models can be obtained considering higher order Taylor series expansion. Using the system model and expanding each state variable up to an order such that the input explicitly appears. This procedure leads to a sampled-data model where the effect of the input appears in all state variables after one sampling interval such as it is in the exact sampled-data model.

In particular, for the induction machine model presented in (3.26)-(3.27), in the stator current state the input explicitly appears. However, for rotor flux we need to perform a second order truncated Taylor series expansion,

$$x_{k+1} = x_k + T_s \left.\dot{x}\right|_k + \frac{T_s^2}{2} \left.\ddot{x}\right|_k . \tag{3.84}$$

Then, using (3.84) and (3.26)-(3.27) the discretized state-space model is derived

$${\mathbf{A_d}}^k{}_{Ta_K} = I + T_s\mathbf{A} + \frac{T_s^2}{2}\mathbf{K}\mathbf{A}^2, \tag{3.85}$$

$${\mathbf{B_d}}^k{}_{Ta_K} = T_s\mathbf{B} + \frac{T_s^2}{2}\mathbf{KAB}, \tag{3.86}$$

$${\mathbf{C_d}}^k{}_{Ta_K} = \mathbf{C}, \tag{3.87}$$

$${\mathbf{D_d}}^k{}_{Ta_K} = \mathbf{D}, \tag{3.88}$$

where Ta_K stands for second order Taylor approximation and the matrix $\mathbf{K}$, is introduced due to the second order Taylor series expansion of the rotor flux, where the effect of the input appears in all state-variables after one sampling interval. The issue related to the relative degree of the resultant discrete-time model is solved by using this method [58].

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.89}$$

3.5.5 Taylor Modified

A modification of the above method is presented here. Now, if $\mathbf{K} = \mathbf{I}$, the second order Taylor series expansion of the system is computed for both states. Then, using (3.84) and (3.26)-(3.27) the new discretized state-space model is derived

$$\mathbf{A_d}^k{}_{Ta_I} = I + T_s\mathbf{A} + \frac{T_s^2}{2}\mathbf{A}^2, \tag{3.90}$$

$$\mathbf{B_d}^k{}_{Ta_I} = T_s\mathbf{B} + \frac{T_s^2}{2}\mathbf{AB}, \tag{3.91}$$

$$\mathbf{C_d}^k{}_{Ta_I} = \mathbf{C}, \tag{3.92}$$

$$\mathbf{D_d}^k{}_{Ta_I} = \mathbf{D}, \tag{3.93}$$

where Ta_I stands for second order Taylor approximation but using $\mathbf{K} = \mathbf{I}$.

3.6 Accuracy Analysis

An error quantification is presented in this section. The objective of this analysis is to try to define a good discretization method for the induction machine model. The analysis is presented for Euler and Matrix Factorization only. However, the error analysis result for all discretization methods is presented in Table 3.2.

3.6.1 Error Quantification for $\mathbf{A_d}^k$

The approximation in $\mathbf{A_d}^k{}_{Eu}$ with respect to exact discretization can be analyzed using (3.46) and a Taylor series expansion of (3.40). This is:

$$\mathcal{E}_{Eu}^{\mathbf{A_d}^k}(T_s) = e^{\mathbf{A}T_s} - (\mathbf{I} + T_s\mathbf{A}) \approx \frac{1}{2!}\mathbf{A}^2T_s^2, \tag{3.94}$$

where, $\mathcal{E}_{EU}^{\mathbf{A_d}^k}(T_s)$ is the error associated to $\mathbf{A_d}^k{}_{Eu}$. Euler approximation is, in fact, a particular case of truncated Taylor series expansion, omitting the second and higher order terms. Notice that the error in $\mathbf{A_d}^k{}_{Eu}$ is quadratic with respect to the speed ω^k and proportional to T_s^2 [62].

Despite the fact that equation (3.62) does not in general hold, it can be used as a good approximation. Experimental results presented in [28, 64, 65] corroborate this fact. The error of the approximate model described in (3.62) can be modeled as

$$\mathbf{A_d}^k = e^{\mathbf{A_c}T_s}e^{\mathbf{A}_\omega T_s} + \mathcal{E}_{MF}^{\mathbf{A_d}^k}(T_s), \tag{3.95}$$

where $\mathcal{E}_{MF}^{\mathbf{A_d}^k}(T_s)$ is the approximation error of $\mathbf{A_d}^k{}_{MF}$ with respect to the exact representation in (3.40). To evaluate this error, the matrix exponential of $\mathbf{A}$, $\mathbf{A_c}$ and $\mathbf{A}_\omega$ can be expressed in Taylor series. Then, we have that

$$\begin{aligned}\mathcal{E}_{MF}^{\mathbf{A_d}^k}(T_s) &= e^{\mathbf{A}T_s} - e^{\mathbf{A_c}T_s}e^{\mathbf{A}_\omega T_s},\\ &= \sum_{k=0}^{\infty}\frac{1}{k!}(\mathbf{A}T_s)^k - \sum_{k=0}^{\infty}\frac{1}{k!}(\mathbf{A_c}T_s)^k\sum_{k=0}^{\infty}\frac{1}{k!}(\mathbf{A}_\omega T_s)^k. \end{aligned}\tag{3.96}$$

Expanding the series in (3.96) up to second order terms in T_s, and then grouping powers of T_s using (3.59), we can truncate the series associated to $\mathcal{E}_{MF}^{\mathbf{A_d}^k}(T_s)$. Then, after extensive algebraic manipulation it is possible to obtain

$$\mathcal{E}_{MF}^{\mathbf{A_d}^k}(T_s) \approx -\frac{1}{2!}(\mathbf{A_c}\mathbf{A}_\omega - \mathbf{A}_\omega\mathbf{A_c})T_s^2. \tag{3.97}$$

Therefore, $\mathcal{E}_{MF}^{\mathbf{A_d}^k}(T_s) \in \mathcal{O}(T_s^2)$, i.e., it is also a function of the order of T_s^2. Then, from (3.97), the approximation error is proportional to T_s^2 and to the difference $\mathbf{A_c}\mathbf{A}_\omega - \mathbf{A}_\omega\mathbf{A_c}$. It can be seen that if the matrices conmute (i.e., $[\mathbf{A_c}, \mathbf{A}_\omega] = 0$, see commutator operator [67]) this discretization strategy would lead to an exact discrete-time model. However, for the induction machine model

$$\mathbf{A_c}\mathbf{A}_\omega - \mathbf{A}_\omega\mathbf{A_c} = \omega^k \begin{bmatrix} 0 & -\frac{L_m k_r}{R_\sigma \tau_r \tau_\sigma} & 0 & -\frac{k_r}{R_\sigma \tau_\sigma^2} \\ \frac{L_m k_r}{R_\sigma \tau_r \tau_\sigma} & 0 & \frac{k_r}{R_\sigma \tau_\sigma^2} & 0 \\ 0 & \frac{L_m}{\tau_r} & 0 & \frac{L_m k_r}{R_\sigma \tau_r \tau_\sigma} \\ -\frac{L_m}{\tau_r} & 0 & -\frac{L_m k_r}{R_\sigma \tau_r \tau_\sigma} & 0 \end{bmatrix}, \tag{3.98}$$

which shows that the approximation error grows linearly with respect to the speed ω^k.

3.6.2 Error Quantification for $\mathbf{B_d}^k$

The error associated to ${\mathbf{B_d}^k}_{Eu}$ can be analyzed using (3.41) and (3.47), to obtain

$$\mathcal{E}_{Eu}^{\mathbf{B_d}^k}(T_s) \approx \frac{1}{2!}\mathbf{A}\mathbf{B}T_s^2. \tag{3.99}$$

From (3.99) it can be noticed that, the error associated to ${\mathbf{B_d}^k}_{Eu}$ is constant with respect to the speed ω^k and proportional to T_s^2. Therefore, when using a Euler approximation, the errors $\mathcal{E}_{Eu}^{\mathbf{A_d}^k}(T_s)$ and $\mathcal{E}_{Eu}^{\mathbf{B_d}^k}(T_s)$ are functions of the order of T_s^2 [62].

Following the same idea above, the error associated to the approximate model described in (3.64) can be expressed as

$$\mathbf{B_d}^k = \int_0^{T_s} e^{\mathbf{A_c}\eta} e^{\mathbf{A}_\omega \eta}\mathbf{B}d\eta + \mathcal{E}_{MF}^{\mathbf{B_d}^k}(T_s), \tag{3.100}$$

where $\mathcal{E}_{MF}^{\mathbf{B_d}^k}(T_s)$ is the approximation error in ${\mathbf{B_d}^k}_{MF}$ with respect to exact representation presented in (3.41). Then,

$$\mathcal{E}_{MF}^{\mathbf{B_d}^k}(T_s) \approx -\frac{1}{3!}(\mathbf{A_c}\mathbf{A}_\omega - \mathbf{A}_\omega\mathbf{A_c})\mathbf{B}T_s^3. \tag{3.101}$$

It can be noticed that in (3.101) the approximation error associated to ${\mathbf{B_d}^k}_{MF}$ when using the model proposed in [28] is proportional to T_s^3 and to the difference $\mathbf{A_c}\mathbf{A}_\omega - \mathbf{A}_\omega\mathbf{A_c}$, therefore $\mathcal{E}_{MF}^{\mathbf{B_d}^k}(T_s) \in \mathcal{O}(T_s^3)$. This shows that the approximation error for ${\mathbf{B_d}^k}_{MF}$ is one order lower than Euler in terms of the sampling period T_s. Furthermore, it can be verified that using (3.67) all the state components are modified by the input in one sampling time (i.e., the model has relative degree one from input to each state). This is what one would expect for an accurate sampled-data description when the system has no pure time delays. This is a key issue, e.g., for finite control set model predictive control as discussed in [58].

Table 3.2: Error quantification of state matrices using different discretization methods.

Matrix Method	$\mathbf{A_d}^k$	$\mathbf{B_d}^k$	$\mathbf{C_d}^k$	$\mathbf{D_d}^k$
Euler	$\frac{T_s^2}{2!}\mathbf{A}^2$	$\frac{T_s^2}{2!}\mathbf{AB}$	$\mathbf{0}$	$\mathbf{0}$
Taylor$_K$	$\frac{T_s^2}{2!}(\mathbf{I}–\mathbf{K})\mathbf{A}^2$	$\frac{T_s^2}{2!}(\mathbf{I}–\mathbf{K})\mathbf{AB}$	$\mathbf{0}$	$\mathbf{0}$
Taylor$_I$	$\frac{T_s^3}{3!}\mathbf{A}^3$	$\frac{T_s^3}{3!}\mathbf{A}^2\mathbf{B}$	$\mathbf{0}$	$\mathbf{0}$
Tustin	$-\frac{T_s^3}{12}\mathbf{A}^3$	$-\frac{T_s^3}{12}\mathbf{A}^2\mathbf{B}$	$-\frac{T_s}{2!}\mathbf{CA}$	$-\frac{T_s}{2!}\mathbf{CB}$
Matrix Fact.	$-\frac{T_s^2}{2!}[\mathbf{A_c}, \mathbf{A}_\omega]$	$-\frac{T_s^3}{3!}[\mathbf{A_c}, \mathbf{A}_\omega]\mathbf{B}$	$\mathbf{0}$	$\mathbf{0}$

Table 3.3: Induction machine parameters

Parameter	Value	Parameter	Value
R_s	1.6647 (Ω)	p	2
R_r	1.2134 (Ω)	J	0.02693 (kgm^2/s)
L_s	136.82 (mH)	ω_{nom}	1440 (rpm)
L_r	136.82 (mH)	T_{nom}	25 (Nm)
L_m	130.69 (mH)	P_{nom}	4.0 (kW)

3.6.3 Error Comparison

A numerical comparison of the approximation errors is presented in this subsection. The proposed approximate sampled-data models are compared with exact discretization using MATLAB to determine a good discretization scheme. The error analysis is based in two norms: Euclidean Norm and $\mathcal{H}_2$ Norm. The induction machine parameters used are presented in Table 3.3. The sampling time is $T_s = 40(\mu\text{s})$ and it was selected according a tradeoff between simulation and hardware platforms. Analysis with different sampling times are presented in [62].

3.6.3.1 Euclidean Norm

For error evaluation the $||\cdot||_2$ norm is used (Euclidean norm [67]). The norm $||\cdot||_2$ of a matrix $\mathbf{M}$ is defined as

$$||\mathbf{M}||_2 = \sqrt{\lambda_{max}(\mathbf{M}^T\mathbf{M})}, \tag{3.102}$$

where λ_{max} is the maximum eigenvalue.

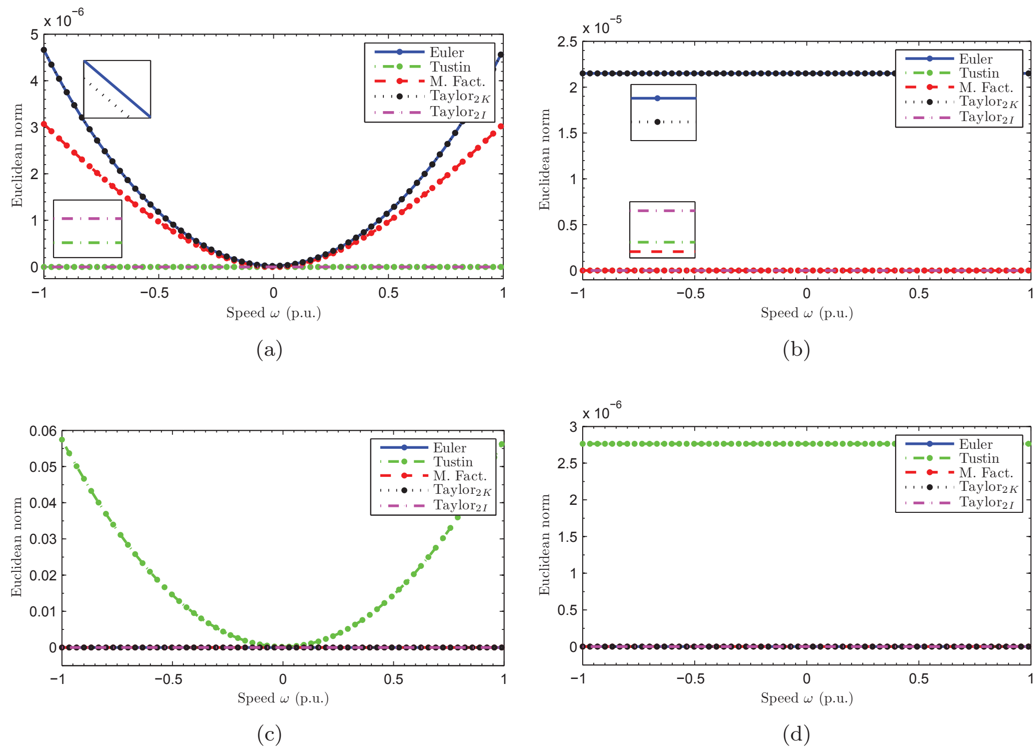


Figure 3.2: Comparison between the normalized error of the exact discrete-time and approximation system using Euclidean-norm for: (a) $\mathcal{E}^{\mathbf{A_d}^k}(T_s)$; (b) $\mathcal{E}^{\mathbf{B_d}^k}(T_s)$; (c) $\mathcal{E}^{\mathbf{C_d}^k}(T_s)$; (d) $\mathcal{E}^{\mathbf{D_d}^k}(T_s)$.

Now, if all terms for each discretization method in $\mathcal{E}^{\mathbf{A_d}^k}(T_s)$, $\mathcal{E}^{\mathbf{B_d}^k}(T_s)$, $\mathcal{E}^{\mathbf{C_d}^k}(T_s)$ and $\mathcal{E}^{\mathbf{D_d}^k}(T_s)$ are considered, the normalized error of the approximate discrete-time matrix ($\tilde{\mathbf{M}}$) compared with the exact discretization ($\mathbf{M}$) can be obtained as follows

$$\boldsymbol{\Delta}\mathbf{M} = \frac{||\mathbf{M} - \tilde{\mathbf{M}}||_2}{||\mathbf{M}||_2}. \tag{3.103}$$

Fig. 3.2a, Fig. 3.2b, Fig. 3.2c and Fig. 3.2d show the comparison of the approximation errors for state matrices $\mathbf{A_d}^k$, $\mathbf{B_d}^k$, $\mathbf{C_d}^k$, $\mathbf{D_d}^k$, respectively. The errors are calculated using a sampling time of $T_s = 40(\mu\text{s})$ and for the full-speed range.

Fig. 3.2a and Fig. 3.2b show that the normalized errors using Matrix Factorization method are smaller than the errors obtained when using the Euler approximation, and this holds for the complete speed range. In particular, the input matrix shows significantly less error (three orders of magnitude for the value of T_s considered). Then, the discretization proposed in [28] (Matrix Factorization) clearly achieves a better behavior in terms of errors than Euler approximation.

Fig. 3.2c and Fig. 3.2d show that the normalized errors using Tustin Discretization are bigger than the error obtained when using other discretization methods. Furthermore, the implementation of this method makes it a bad alternative in terms of computation time.

From Fig. 3.2b Taylor modified give better results than Taylor, but its implementation is a bit more complicate than Taylor presented in [58].

The Euclidean norm evaluation presents results for Matrix Factorization, Taylor and Taylor Modified. However, the analysis is performed for each state matrix. This is a drawback for the designer as a more general norm for the system error is needed. A new norm involving all the system matrices has been studied: the $\mathcal{H}_2$-norm.

The discrete-time model used in [28] for an induction machine is only an approximate sampled-data model. Mathematical background and an accuracy analysis have been provided. Finally, the numerical analysis presented in this section shows that, the discrete-time model proposed in [28] provides an accurate approximation, in particular, when compared to the usual Euler method. The presented analysis provides a theoretical basis to experimental implementations.

3.6.3.2 $\mathcal{H}_2$ Norm

3.6.3.2.1 Continuous-Time Systems The $\mathcal{H}_2$-norm is based on observability and controllability Gramian. The observability Gramian of a stable linear system with state matrix $\mathbf{A}$ and observation matrix $\mathbf{C}$ can be written as

$$\mathbf{W_o} = \int_0^{\infty} e^{\mathbf{A}^T t}\mathbf{C}^T\mathbf{C}e^{\mathbf{A}t}dt. \tag{3.104}$$

Similarly, the controllability Gramian of a stable linear system with state matrix $\mathbf{A}$ and input matrix $\mathbf{B}$ can be written as

$$\mathbf{W_c} = \int_0^{\infty} e^{\mathbf{A}^t}\mathbf{B}\mathbf{B}^T e^{\mathbf{A}^T t}dt. \tag{3.105}$$

$\mathbf{W_c}$ and $\mathbf{W_o}$ give information about observability and controllability of the system. Controllability and observability Gramian are computed by solving the Lyapunov equations [67]:

$$\mathbf{A}\mathbf{W_c} + \mathbf{W_c}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = \mathbf{0}, \tag{3.106}$$

$$\mathbf{A}^T\mathbf{W_o} + \mathbf{W_o}\mathbf{A}^T + \mathbf{C}^T\mathbf{C} = \mathbf{0}. \tag{3.107}$$

Then, the $\mathcal{H}_2$ norm of a continuous-time system matrices can be expressed in terms of the Gramians as

$$||\cdot||_{\mathcal{H}_2} = \sqrt{tr(\mathbf{B}^T\mathbf{W_o}\mathbf{B})} = \sqrt{tr(\mathbf{C}\mathbf{W_c}\mathbf{C}^T)}. \tag{3.108}$$

3.6.3.2.2 Discrete-Time Systems Now, for a discrete-time system, controllability and observability Gramian are computed by solving the Lyapunov equations:

$$\mathbf{A_d}\mathbf{W_{cd}}\mathbf{A}^T - \mathbf{W_{cd}} + \mathbf{B_d}\mathbf{B_d}^T = \mathbf{0}, \tag{3.109}$$

$$\mathbf{A_d}^T\mathbf{W_{od}} - \mathbf{W_{od}} + \mathbf{C_d}^T\mathbf{C_d} = \mathbf{0}. \tag{3.110}$$

where controllability and observability Gramian [68, 69] are defined as

$$\mathbf{W_{od}}(0,k) = \sum_{j=0}^{k-1} \mathbf{A_d}^{Tj}\mathbf{C_d}^T\mathbf{C_d}\mathbf{A_d}^j. \tag{3.111}$$

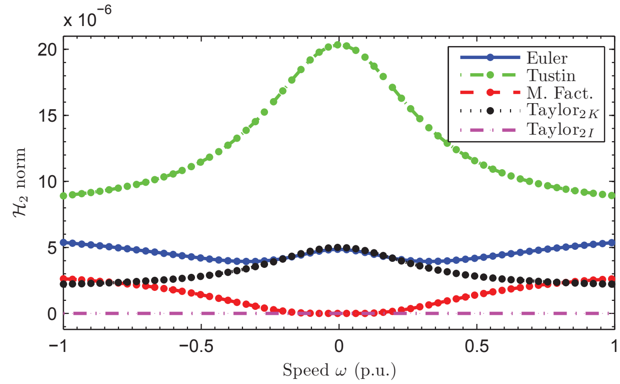


Figure 3.3: Comparison between the normalized error of the exact discrete-time and approximation system using $\mathcal{H}_2$-norm.

$$\mathbf{W_{cd}}(0,k) = \sum_{j=0}^{k-1} \mathbf{A_d}^j \mathbf{B_d} \mathbf{B_d}^T \mathbf{A_d}^{Tj}. \tag{3.112}$$

Then, the $\mathcal{H}_2$-norm of a discrete-time system matrices can be expressed in terms of the Gramians as

$$|| \cdot ||_{\mathcal{H}_2} = \sqrt{tr(\mathbf{B_d}^T \mathbf{W_{od}} \mathbf{B_d})} = \sqrt{tr(\mathbf{C_d} \mathbf{W_{cd}} \mathbf{C_d}^T)}. \tag{3.113}$$

Now, using $\mathcal{H}_2$ based on the observability-Gramian based (OGB) approach, the Comparison between the normalized error of the exact representation and its approximation is presented in Fig. 3.3. From the above, errors using Tustin and Euler methods are more higher than Taylor and Matrix Factorization approximations. Finally, Taylor Modified and Matrix Factorization are good candidates for accurate discretization methods.

3.7 Conclusions

This chapter clarifies the accuracy of the discrete-time model of an induction machine by using Euler, Tustin, Taylor and Matrix Factorization approximations. The studied methods correspond, in fact, to an approximate sampled-data model. Mathematical background and an accuracy analysis have been provided for each case.

The numerical analysis presented in this chapter shows that, the discrete-time model proposed in [28] provides an accurate approximation, in particular, when compared to the usual Euler method. Matrix factorization, Taylor and Taylor modified are good alternatives to used in experimental test bench. Instead of results obtained with Tustin are good, this approximation has two main drawbacks, the first is the computational burden due to it involve matrix inversion and matrix multiplication, which increase the sampling time. Finally, the best discretization model in terms of numerical error is achieved with the modified Taylor method.

Chapter 4

PREDICTIVE TORQUE AND FLUX CONTROL (PTC)

4.1 Introduction

THE standard PTC approach uses a single cost function built by a linear combination of the objective functions, to determine the best voltage vector to apply in the next sampling time. The torque and the flux errors are included in one cost function through the use of weighting factors [28,32,34]. These factors depend on the operating point and system parameters [41], so their choice is not a trivial task. Furthermore, the weighting factors have influence in the performance of the controller because they determine the relative importance of the torque and stator flux. The implementation of PTC depends strongly on the *System Model*, for this reason this topic has been studied in the previous chapter. In this new chapter the conventional PTC scheme is presented. The problem of weighting factors is studied in depth. Finally, a Predictive Current Control is introduced as a kind of an alternative to PTC. Finally, the algorithm is analyzed with several simulations in different operation points.

4.2 Conventional PTC Scheme

As in DTC, stator flux $\mathbf{\Psi_s}$ and electromagnetic torque T can be modified by selecting a proper voltage vector $\mathbf{v_s}$, which modifies the magnitude of the stator flux and at the same time increases or decreases the angle between the rotor and stator flux. These ideas correspond to the basics of Direct Torque Control (DTC). In PTC, the same principle is used, but in this scheme, predictions for the future values of the stator flux and torque are computed. Hence, the reference condition, which is implemented by a cost function, considers the future behavior of these variables. The predictions are calculated for every actuating possibility and the cost function selects the voltage vector which optimizes the reference tracking.

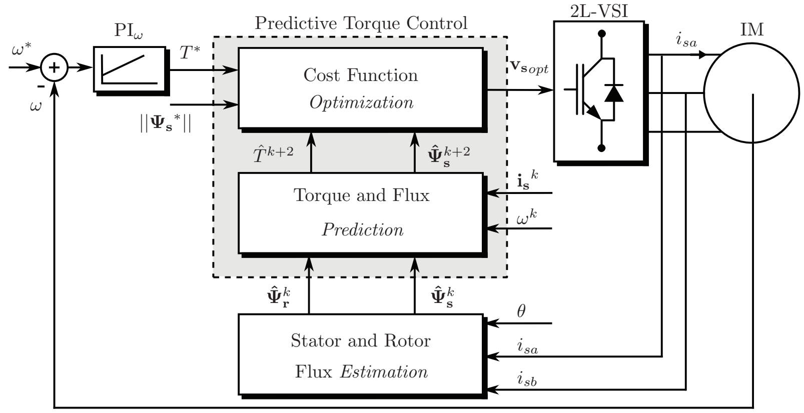


Figure 4.1: Predictive torque and flux control diagram.

4.2.1 PTC Control Diagram

The execution of the PTC algorithm can be divided in three main steps: *Estimation* of the variables that cannot be measured, *Prediction* of the future plant behavior and *Optimization* of the single cost function according to a reference condition. The PTC scheme is shown in Fig. 4.1, with $\mathbf{v_s}$ the stator voltage vector, $\mathbf{i_s} = [i_{sa}\ i_{sb}\ i_{sc}]^T$ the measured stator current vector, $\mathbf{i_r}$ being the rotor current vector, $\hat{\mathbf{\Psi}}_\mathbf{s}$ being the estimated stator flux vector, $\hat{\mathbf{\Psi}}_\mathbf{r}$ being the estimated rotor flux vector, $\hat{T}^*$ being the estimated electrical torque reference, ω being the measured mechanical speed and ω^* its reference. Finally, $\mathbf{v_s}_{opt}$ is the optimal voltage vector to apply in the next sampling time.

The block called *Estimation* is used to compute the current values of the variables that cannot be measured, as the rotor flux $\hat{\mathbf{\Psi}}_\mathbf{r}$ and the stator flux $\hat{\mathbf{\Psi}}_\mathbf{s}$. The predictive model computes the future values of controlled variables in the instant $k+2$, in this case the stator flux $\hat{\mathbf{\Psi}}_\mathbf{s}^{k+2}$ and the electromagnetic torque $\hat{T}^{k+2}$. These predictions are calculated for every actuating possibility given by the inverter topology. If a 2L-VSI inverter is considered, eight switching states and seven different voltage vectors can be generated. The block called *Optimization* chooses the optimum switching state, which minimizes the corresponding cost function. The function contains the control law to reach the torque and stator flux references according to the references.

4.2.2 Speed Controller

The drive considers a cascade control loop, composed by a non-linear internal controller (torque and flux control) and an external PI-speed controller (PI$_\omega$), where the speed control-loop bandwidth is lower than the inner loop. Remember that speed is a mechanical variable, it is slower than electrical variables. Furthermore, FCS-MPC has a high bandwidth allowing a high decoupling degree between inner and outer loop [34]. The output of PI$_\omega$-controller corresponds to the torque reference T^* for the predictive inner loop.

The above indicates that the speed dynamic has been imposed by the PI$_\omega$ bandwidth. If a faster response is needed, e.g., for servomotors or PMSM drive applications, one possibility is to use a Predictive Speed Control (PSC), where the speed control is included in the same cost function within torque and flux. This idea has been widely developed in the literature with linear and non-linear controller, but it is not part of the proposed investigation [25, 70].

The speed controller is designed by using the speed equation of the machine.

$$J\frac{d\omega}{dt} = T - T_l. \tag{4.1}$$

Now, assuming T_l as an external disturbance and applying Laplace transform to (4.1),

$$\omega = \frac{1}{Js}T. \tag{4.2}$$

Then, is well-known that under a rotating frame in alignment with the rotor flux angle, the electromagnetic torque T is proportional to the rotor flux and the complex component of the stator current,

$$T = \frac{3}{2}\frac{L_m}{L_r}p\psi_{rd}i_{sq}. \tag{4.3}$$

The result of (4.3) means that under a rotating reference frame, the electromagnetic torque T is proportional to the rotor flux ψ_{rd} and the quadrature component of the stator current i_{sq}. Due to that the rotor flux is controlled to a constant value by the direct component of the stator current i_{sd}, the electromagnetic torque can be commanded only by i_{sq}. Finally, the continuous plant of the speed loop is obtained,

$$\frac{\omega}{i_{sq}} = \frac{3L_m p}{2L_r J}\frac{1}{s}. \tag{4.4}$$

Finally, the external speed controller designed in discrete-time as

$$\mathrm{PI}_\omega(z) = \frac{k_{p\omega} + k_{i\omega}z^{-1}}{1 - z^{-1}}, \tag{4.5}$$

and implemented with anti-windup in order to limit the value of i_{sq} and then of the torque reference [71]. The electromagnetic torque reference T is implemented by using (4.6),

$$T^* = \frac{3}{2}\frac{L_m}{L_r}p\psi_{rd}i^*_{sq}. \tag{4.6}$$

Thus, the saturation value of i^*_{sq} is calculated using nominal torque value T_{nom} for a given ψ_{rd}. This value is calculated solving (4.7)-(4.10) for ψ_{rd},

$$T_{nom} = \frac{3}{2}\frac{L_m}{L_r}p\psi_{rd}i_{sq}, \tag{4.7}$$

$$\psi_{sd} = \sigma L s i_{sd} + k_r\psi_{rd}, \tag{4.8}$$

$$\psi_{sq} = \sigma L s i_{sq}, \tag{4.9}$$

$$\psi_{rd} = L_m i_{sd}, \tag{4.10}$$

where (4.8)-(4.10) are derived by using (3.8)-(3.9). Now, considering that the module of stator flux is $\psi^2_{s_{nom}} = \psi^2_{sd} + \psi^2_{sq}$, the value of ψ_{rd} is calculated by solving (4.11),

$$\psi_{s_{nom}}^2 = \left(\frac{L_s \psi_{rd}}{L_m}\right)^2 + \left(\frac{2\sigma LsT_{nom}}{3pk_r\psi_{rd}}\right)^2, \tag{4.11}$$

Then, for example, if $\psi_{s_{nom}} = 0.980$ (Wb) and $T_{nom} = 25.0$ (Nm), the value of ψ_{rd} is 0.930 (Wb). Finally, saturation value of i_{sq}^* is calculated resolving (4.7)-(4.10) for $\psi_{rd} = 0.930$ (Wb). Thus, saturation value of i_{sq}^* is $i_{sq}^{sat} = 9.381$ (A) and magnetization current is $i_{sd}^{mag} = 7.115$ (A).

4.2.3 Flux Estimation

One of the most important challenges in every electrical drive, is the rotor or stator flux estimation, depending on which control strategy is implemented. In the case of rotor flux orientation, this method is based on obtaining the rotor flux angle θ_s in order to express all the electromagnetic variables in a synchronous rotating coordinate frame (dq-frame). The rotor angle θ_s can be computed using (4.12),

$$\hat{\theta}_s = \arctan\left(\frac{\hat{\psi}_{r\beta}}{\hat{\psi}_{r\alpha}}\right). \tag{4.12}$$

In order to estimate the non-measurable variables, the implementation of estimators is based on a discrete-time model of the machine and measurable variables, such as stator current, voltage and rotor speed. Basically, estimators can be divided in open-loop and closed-loop estimators. Closed-loop estimators are well-known as observers, they present a predictive correction in order to assure a faster convergence and a better robustness of the estimation under changes of the system parameters. However, estimators without feedback or open-loop estimators are the most simple scheme in terms of implementations. In the same way, estimations without feedback can be separated in two branches, direct methods are based on the use of the stator voltage and current measurements, and indirect methods, which utilize an inherit slip relation. The direct method using current measurements is one of the most conventional and well-known rotor flux estimators. The main advantage is its simple implementation and numerical convergente. For this reason, this method has been used in this work.

In PTC [28, 34], estimations of the stator flux $\mathbf{\Psi_s}$ and the rotor flux $\mathbf{\Psi_r}$ are required at the present sampling time k. As in FOC [6], the rotor flux can be calculated using the equivalent equation of the rotor dynamics of a squirrel-cage induction machine in rotating reference frame aligned with the rotor winding which gives:

$$\mathbf{\Psi_r} + \tau_r \frac{d\mathbf{\Psi_r}}{dt} = L_m \mathbf{i_s}, \tag{4.13}$$

where, $\tau_r = \frac{L_r}{R_r}$ is the rotor time constant and T_s the sampling time. Applying Laplace transform to (4.13), the rotor flux estimation is obtained:

$$\mathbf{\Psi_r} = \frac{L_m}{\tau_r s + 1} \mathbf{i_s}. \tag{4.14}$$

The corresponding block diagram is illustrated in Fig. 4.2. Using the discretization

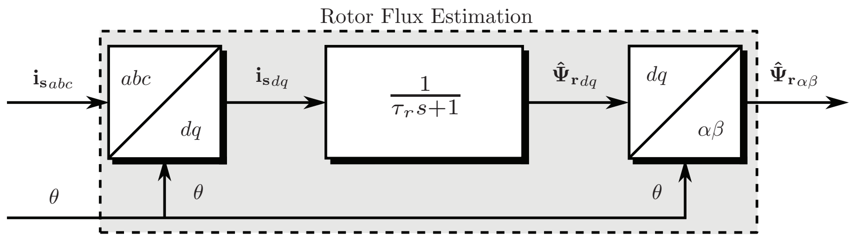


Figure 4.2: Rotor flux estimation based on stator current in rotating coordinates.

presented in (3.40)-(3.43), the discrete-time equation for the rotor flux estimation is:

$$\hat{\mathbf{\Psi}}_{\mathbf{r}dq}^{k} = e^{\frac{-T_s}{\tau_r}}\hat{\mathbf{\Psi}}_{\mathbf{r}dq}^{k-1} + L_m\left(1 - e^{\frac{-T_s}{\tau_r}}\right)\mathbf{i_s}_{dq}^{k-1}. \tag{4.15}$$

Finally, the rotor flux in a stationary frame is computed as

$$\hat{\mathbf{\Psi}}_{\mathbf{r}\alpha\beta}^{k} = [\hat{\mathbf{\Psi}}_{\mathbf{r}\alpha}^{k}\ \hat{\mathbf{\Psi}}_{\mathbf{r}\beta}^{k}]^T, \tag{4.16}$$

$$\hat{\mathbf{\Psi}}_{\mathbf{r}\alpha}^{k} = \hat{\mathbf{\Psi}}_{\mathbf{r}d}^{k} cos(\theta^k) - \hat{\mathbf{\Psi}}_{\mathbf{r}q}^{k} sin(\theta^k), \tag{4.17}$$

$$\hat{\mathbf{\Psi}}_{\mathbf{r}\beta}^{k} = \hat{\mathbf{\Psi}}_{\mathbf{r}d}^{k} sin(\theta^k) + \hat{\mathbf{\Psi}}_{\mathbf{r}q}^{k} cos(\theta^k), \tag{4.18}$$

with θ^k the sampled value of the rotor angular position given directly by the encoder. Now, the stator flux can be estimated by replacing $\mathbf{i_r}_{\alpha\beta}$ of (3.15) in the stator flux linkage equation (3.16). Thus, the stator flux at the present sampling time k is

$$\hat{\mathbf{\Psi}}_{\mathbf{s}\alpha\beta}^{k} = k_r\hat{\mathbf{\Psi}}_{\mathbf{r}\alpha\beta}^{k} + \sigma L_s \mathbf{i_s}_{\alpha\beta}^{k}, \tag{4.19}$$

where $k_r = \frac{L_m}{L_r}$ is the rotor coupling factor and $\sigma = 1 - \frac{L_m^2}{L_s L_r}$ is the total leakage factor.

4.2.4 Predictions

Since the control variables used in PTC are the stator flux and the electromagnetic torque, their behavior must be predicted at the sampling step $k+1$. The stator flux prediction $\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+1}$ is obtained using the discrete-time model presented in (3.33)-(3.34). Finally, the expression for $\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+1}$ and $\mathbf{i_s}^{k+1}$ are determined by the used discretization approximation.

$$\begin{bmatrix}\mathbf{i_s}^{k+1}\\ \hat{\mathbf{\Psi}}_{\mathbf{r}}^{k+1}\end{bmatrix} = \mathbf{A_d}^k\begin{bmatrix}\mathbf{i_s}^{k}\\ \hat{\mathbf{\Psi}}_{\mathbf{r}}^{k}\end{bmatrix} + \mathbf{B_d}^k\left[\mathbf{v_s}^k\right], \tag{4.20}$$

$$\begin{bmatrix}\mathbf{i_s}^{k+1}\\ \hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+1}\end{bmatrix} = \mathbf{C_d}^k\begin{bmatrix}\mathbf{i_s}^{k+1}\\ \hat{\mathbf{\Psi}}_{\mathbf{r}}^{k+1}\end{bmatrix} + \mathbf{D_d}^k\left[\mathbf{v_s}^{k+1}\right]. \tag{4.21}$$

The electromagnetic torque prediction $\hat{T}^{k+1}$ depends on the stator flux $\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+1}$ and stator current $\mathbf{i_s}^{k+1}$ predictions according to (3.10):

$$\hat{T}^{k+1} = \frac{3}{2}p\left(\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+1} \times \mathbf{i_s}^{k+1}\right). \tag{4.22}$$

In a real-time applications the time required to compute the control algorithm takes a significant portion of the sample period, resulting in one sampling delay [72]. The effect of this delay has a large impact on the controller performance, therefore a delay compensation scheme must be implemented [34, 59, 73]. Here, a model-based prediction is used to compensate the calculation delay, where the variables at the next sampling period $k+1$ are extrapolations used as an initial condition for the prediction at the instant $k+2$. In the case of torque and flux the predictions at the sampling period $k+2$ are calculated according to (4.21) and (4.22) but shifted one sample step to predict $\hat{T}^{k+2}$ and $\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}$. Then,

$$\begin{bmatrix} \mathbf{i_s}^{k+2} \\ \hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2} \end{bmatrix} = \mathbf{C_d}^k \begin{bmatrix} \mathbf{i_s}^{k+2} \\ \hat{\mathbf{\Psi}}_{\mathbf{r}}^{k+2} \end{bmatrix} + \mathbf{D_d}^k \left[\mathbf{v_s}^{k+2} \right], \tag{4.23}$$

$$\hat{T}^{k+2} = \frac{3}{2} p \left(\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2} \times \mathbf{i_s}^{k+2} \right). \tag{4.24}$$

where, the variables $\hat{T}^{k+2}$ and $\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}$ depend on the next voltage vectors $\mathbf{v_s}^{k+1}$ only due to $\mathbf{D_d}^k = \mathbf{0}$. In the case of the 2L-VSI, the valid voltage vectors are 8, $\{\mathbf{v_0}, \mathbf{v_1}, ..., \mathbf{v_7}\}$ [18].

4.2.5 Optimization

The next stage considers the minimization of a single objective function that pursues torque and flux tracking at every sampling time T_s. Then, considering the elimination of the calculation delay in digital implementation, the cost function to minimize has the following structure:

$$G = k_1 g_1 + k_2 g_2, \tag{4.25}$$

$$= k_T g_T + k_\Psi g_\Psi \tag{4.26}$$

$$= k_T \left(T^* - \hat{T}^{k+2} \right)^2 + k_\Psi \left(||\mathbf{\Psi_s}^*|| - ||\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}|| \right)^2 \tag{4.27}$$

where, the variables $\hat{T}^{k+1}$ and $\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}$ are calculated using (4.23) and (4.24). Note that, they depend on the voltage vector $\mathbf{v_s}^{k+1}$. The torque reference T^* is externally generated by a PI-speed controller and $||\mathbf{\Psi_s}^*||$ is the stator flux reference. Thus, the stator voltage vector that minimizes (4.27) is selected to apply at the next sampling time,

$$\mathbf{v_s}_{opt} = \arg \min_{\{\mathbf{v_0}, ..., \mathbf{v_7}\}} G(\mathbf{v_s}^{k+1}), \tag{4.28}$$

where, $\mathbf{v_s}_{opt}$ is the optimal stator voltage vector to apply in the next sampling time.

The weighting factors of (4.26) are two: k_Ψ and k_T. For simplicity, the weighting factor associated with the torque k_T is considered unity (k_T=1). Thus, the weighting factor to adjust in (4.26) is k_Ψ only, which penalizes the importance of the flux over the torque control. At the present state of the art, this weighting factor is determined analytically [41], with genetic algorithms [43] and empirically [34, 42, 45]. Naturally, the selection of the k_Ψ value is a difficult task and it has a significant influence in the performance of the controller.

4.3 Weighting Factors Determination Problem

The literature up to now has revealed that a key issue in PTC implementations is the selection of the weighting factor used in the cost function. Weighting factors are used to give more importance to one or another variable and to normalize the different control objectives [32]. These scalar factors are parameters to adjust and its selection is an important task, because it is more complex than the tuning of PI coefficients or hysteresis bands of traditional controllers. Several methods using offline and online search procedures have been implemented at the present state of the art, but they strongly depend on system parameters and are only formulated for two control objectives in a specific application and not in a very systematic way [41, 44]. When more objectives are considered, the weighting factors are usually obtained using trial and error procedures and running time-consuming simulations [42, 43, 45].

In PTC, the weighting factor k_Ψ of the cost function presented in (4.27) is the parameter to adjust. A starting point to the weighting factor is given by

$$k_\Psi = \left(\lambda \frac{T_n}{||\mathbf{\Psi_s}_n||} \right)^2 , \tag{4.29}$$

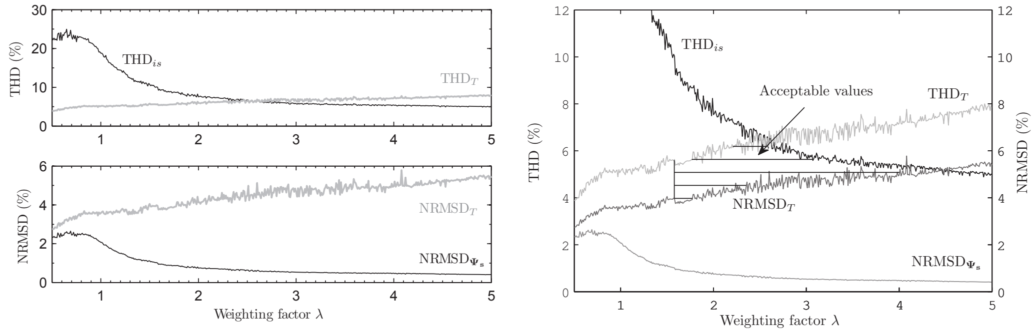
where T_n and $||\mathbf{\Psi_s}_n||$ are the nominal values of the torque and stator flux, respectively. The term λ is currently obtained experimentally by a heuristic procedure. Note that, this fixed weighting factor should be tuned offline for a correct operation through a wide operating range, resulting a complex drive commissioning, [34]. Another solution is presented in [41], where the weighting factor is calculated online in an analytical way. This alternative is strongly system-parameters dependent and requires a comprehensive mathematical analysis. However, in applications where the cost function is composed of variables with the same nature (same units and order of magnitude) or it is a decomposition of a single variable into two components, weighting factors tuning is not necessary [45].

Finally, the problem of weighting factors is addressed in three different ways, the first is based on offline sweep of the k_Ψ in terms of different performance indices. The second method transforms the control problem, from a torque and stator flux control to predictive current control in a rotating frame, where the direct and quadrature components of the stator current will command the magnetization and torque of the machine, respectively. The last proposed method is to avoid completely the use of weighting factors by using multiobjective optimizations. This method is presented in the next chapter.

4.3.1 Offline Parameter Sweep

In PTC, the weight factor of the cost function is the only parameter to adjust, making this feature one of the main advantages of this strategy. However, at the same time, it is also a disadvantage [41]. In this method, the value of the weight factor given by (4.29) has been taken only as a starting point; the final value was obtained by a heuristic procedure through offline time-consuming simulations (parameter sweep).

The simulation settings are illustrated in Table 4.2. The PTC control method is programmed in `C` using MATLAB/Simulink, where the used approximation model is the modified Taylor because it corresponds to the best approximation to the continuous model according to the study presented in the above chapter. The simulation consider 20 periods of the stator current and two merit functions, defined to evaluate the performance of the system working under each set of parameters. The selected performance indices are Total Harmonic Distortion (THD) and Normalized Root-Mean-Square Deviation (NRSMD) of



(a) (b)

Figure 4.3: Simulation sweep of λ in PTC, (a) variation of THD and NRSMD performance indices; (b) acceptable values of λ.

stator current and torque,

$$\mathtt{THD} = 100\sqrt{\frac{\sum_{i=0}^{50} h_i^2}{h_1^2} - 1}, \tag{4.30}$$

$$\mathtt{NRSMD} = 100\sqrt{\frac{\sum_{i=1}^{l}(x_i - \tilde{x})^2}{(l-1)\tilde{x}^2}}, \text{ where } \tilde{x} = \sum_{i=1}^{l} x_i \tag{4.31}$$

where h_i is the i-th harmonic of the stator current and torque; x_i and $\tilde{x}$ are the i-th value of each variable and its mean value, respectively [42]. The result illustrated in Fig. 4.3a is a parameter sweep using 450 simulations. The idea of Fig. 4.3a is the calculation of an optimal weight factor λ_{opt}, which satisfies the design requirements, e.g., $\lambda_{opt} = 2.56$ gives the better conditions for THD of stator current (THD$_{is}$) and torque (THD$_T$). In fact, an acceptable values region can be defined by using all the three performance indices. This region is presented in Fig. 4.3b.

4.3.2 Predictive Current Control (PCC): An Alternative

In PTC, the control variables have different dynamics and magnitudes (torque and flux), thus the associated weight factor is operation-point and system-parameters dependent [41]. One alternative to simplify the weight factor search is to transform the cost function from a torque and stator flux control to predictive current control (PCC) in a rotating frame, where the direct and quadrature components of the stator current will command the magnetization and torque of the machine, respectively. This alternative is called finite control set model predictive field-oriented control. In PCC, the cost function to minimize has the following

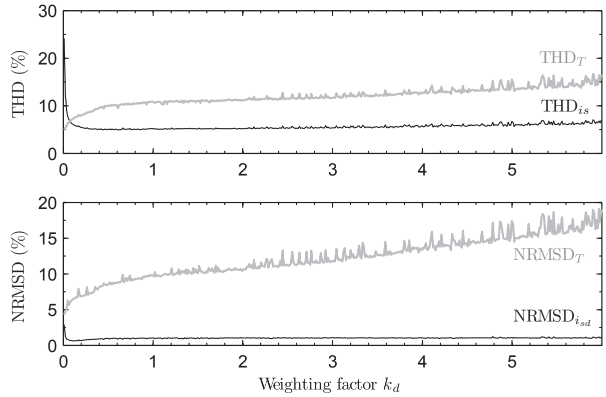


Figure 4.4: Variation of k_d for THD and NRSMD performance indices in the PCC control method.

structure:

$$G = k_1 g_1 + k_2 g_2, \tag{4.32}$$

$$= k_d g_d + k_q g_q \tag{4.33}$$

$$= k_d \left(i_{sd}^* - i_{sd}^{k+2}\right)^2 + k_q \left(i_{sq}^* - i_{sq}^{k+2}\right)^2 \tag{4.34}$$

where, the variables i_{sd}^{k+2} and i_{sq}^{k+2} are calculated using (3.22). The torque reference i_{sq}^* is externally generated by a PI-speed controller and i_{sd}^* is the magnetizing stator current commanded by the user. Thus, the stator voltage vector that minimizes (4.34) is selected to apply at the next sampling time.

The weighting factors of (4.34) are two: k_d and k_q. They are considered unity ($k_d = k_q = 1$) by simplicity. However, the control problem have not the same flexibility that offer PTC. One alternative is to consider $k_q = 1$ and try to adjust k_d, which penalizes the importance of the flux control over the torque [19]. Fig. 4.4 shows a several values of k_d for each performance index. Note that $k_d = 0.61$ gives the better conditions for THD of stator current (THD$_{is}$).

4.4 Control Flexibility

In a control system it is important to have a compromise between reference tracking and control effort. In power converters and drives, the control effort is related with the voltage or current variations, the switching frequency or the switching losses. Using PTC, it is possible to consider some variations in the cost function, in order to reduce control effort. These variations are classified in three groups: implementation of switching schemes, input constraints and output constraints.

4.4.1 Switching Schemes

Three different switching schemes have been implemented. The main idea is to minimize the switching frequency but without increase significantly the THD of current and torque.

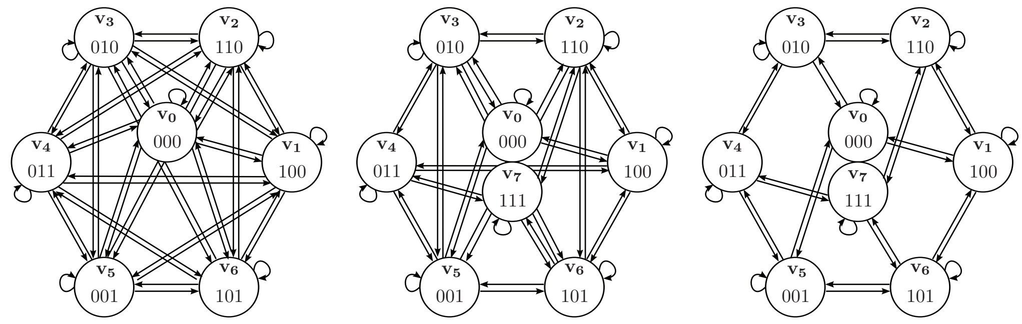


Figure 4.5: Different switching schemes by using: (a) active vectors without zero-redundances; (b) active vectors with zero-redundances; (c) adjacent vectors.

For this reason a tradeoff between switching frequency and THD is required. Fig. 4.5a illustrates a switching scheme by using active vectors without zero-redundances, where only one zero vector is used (e.g., $\mathbf{v_0}$). Fig. 4.5b shows a switching scheme by using active vectors with zero-redundances, where two zero vectors are used ($\mathbf{v_0}$ and $\mathbf{v_7}$). The transitions to zero vectors are restricted to the closest zero vector. Finally, a switching scheme using adjacent vectors is presented in Fig. 4.5c, where two zero vectors are used ($\mathbf{v_0}$ and $\mathbf{v_7}$). The transitions between active and zero vectors are restricted to the closest zero vector [26].

4.4.2 Input Constraints

The second variation to the cost function is the inclusion of input constraints. In the literature, two different input constraints are considered. They are commutations and voltage common-mode reduction.

4.4.2.1 Commutation Number Reduction

As in power converters one of the major measures of control effort is the switching frequency, it is important in many applications to be able to control or limit the number of commutations of the power switches. In order to consider in the cost function the reduction of the number of commutations, a simple approach is to include a term in the cost function that is the number of switches that change when the next voltage vector ${\mathbf{v_s}}^{k+1}$ is applied, with respect to the previously applied switching state ${\mathbf{v_s}}^{k}$. The resulting cost function is expressed as

$$G = k_T \left(T^* - \hat{T}^{k+2}\right)^2 + k_\Psi \left(||{\mathbf{\Psi_s}}^*|| - ||\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}||\right)^2 + k_n n_c, \tag{4.35}$$

where, n_c is the number of switches that change when the next voltage vector ${\mathbf{v_s}}^{k+1}$ is applied and k_n is the weight factor associated to the commutation number. Then, n_c is derived as

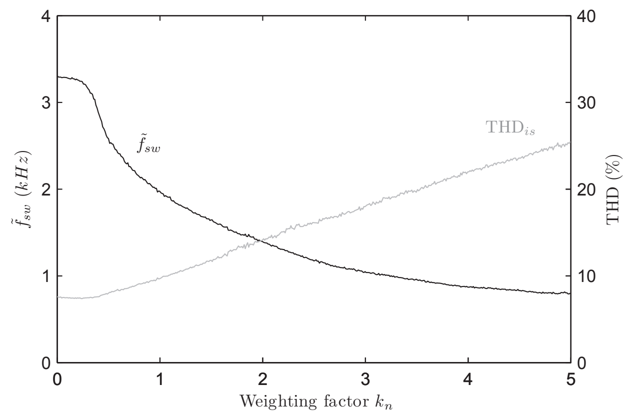


Figure 4.6: Variation of the switching frequency in the PTC control method.

Table 4.1: Some merit functions by using different switching schemes

Switching schemes	THD$_{is}$ (%)	$\tilde{f}_{sw}$ (kHz)
Active vectors without zero-redundances	7.61	3.31
Active vectors with zero-redundances	7.61	3.29
Adjacent vectors	7.71	3.26
Commutations reduction with k_n =1.97	13.67	1.42

$$n_c = \sum_{i=a,b,c} |\mathbf{v_s}^{k+1}(S_i) - \mathbf{v_s}^{k}(S_i)|, \tag{4.36}$$

where $\mathbf{v_s}^{k+1}(S_i)$ and $\mathbf{v_s}^{k}(S_i)$ represent the switching state of phase i, $(i = a, b, c)$ at the present cycle (during k-th and $(k+1)$-th sampling instant) and the next cycle (during $(k+1)$th and $(k+2)$th sampling instant), respectively [74]. For example, if $\mathbf{v_s}^k = \mathbf{v_1}(100)$, then $\mathbf{v_s}^k(S_a) = 1$, $\mathbf{v_s}^k(S_b) = 0$ and $\mathbf{v_s}^k(S_c) = 0$.

Thus, considering the new cost function presented in (4.35) there are two adjustable weighting factors, further the weighting factors selection problem becomes more complex. If n_c is too large, i.e., reducing commutation frequency is more of concern, the performance of torque and flux will be deteriorated. Now, assuming k_Ψ constant, the variation of the switching frequency under k_n is presented in Fig. 4.6.

From Fig. 4.6 is possible to calculate of an optimal weight factor k_n with respect to THD design requirements, in this case $k_n = 1.97$ gives the better conditions for THD of stator current (THD$_{is}$) and number commutations. Table 4.1 shows the THD$_{is}$ and average switching frequency for the above condition.

Table 4.1 shows some merit functions using the switching schemes presented in Fig. 4.5. The simulation settings are illustrated in Table 4.2, but the load torque is set to zero and $\lambda_{opt} = 2.56$. Optimal voltage vectors are presented in Fig. 4.7.

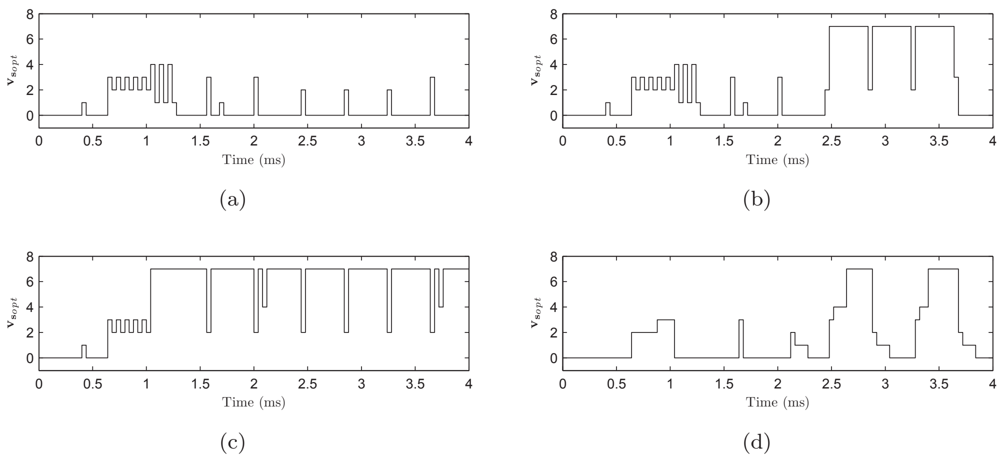


Figure 4.7: Optimal voltage vectors of different switching schemes by using: (a) active vectors without zero-redundances; (b) active vectors with zero-redundances; (c) adjacent vectors; (d) commutations reduction with k_n =1.97.

Another well-known input constraint is the voltage common-mode reduction, which is included in the cost function and minimized with a new weighting factor. However, by including this constraint only the reduction of low-frequency common-mode voltage is achieved.

4.4.3 Output Constraints

In the literature, the more used output constraint is the current limitation. The resulting cost function is expressed as

$$G = k_T \left(T^* - \hat{T}^{k+2}\right)^2 + k_\Psi \left(||\mathbf{\Psi_s}^*|| - ||\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}||\right)^2 + f_c(\mathbf{i_s}^{k+2}), \tag{4.37}$$

with $f_c(\mathbf{i_s}^{k+2})$ defined by

$$f_c(\mathbf{i_s}^{k+2}) = \begin{cases} K, \text{ if } |\mathbf{i_s}| \geq I_s^{max} \\ 0, \text{ if } |\mathbf{i_s}| < I_s^{max}, \end{cases}, \tag{4.38}$$

where K is a very large constant, the function f_c takes a large value when the predicted currents exceed a given limit, acting in practice as constraints on the current magnitudes. Finally, the current limitation is only required when weighting factors are not adjusted (drive commissioning).

4.5 Simulation Results

To validate the proposed PTC scheme, a computer simulation using Matlab/Simulink has been conducted using the parameters given in Table 4.2, which were selected according to the existing parameters of an experimental prototype. The control algorithm presented in Fig. 4.9 has been implemented in `C` because exactly the same code will be used for the experimental tests.

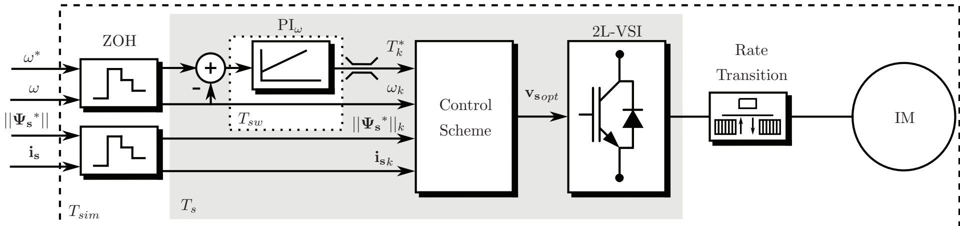


Figure 4.8: Basic simulation scheme.

4.5.1 Configurations

The objective of this section is to simulate the predictive control strategies for electrical drives in order to study their main operation characteristics. These simulations correspond to the control system that utilizes a horizon-one prediction but considering the delay compensation. Basically, this control strategy will be developed with different discrete-time models of the machine, which were presented in the previous chapter. All the simulations are implemented in MATLAB/Simulink and the algorithms are programmed in `C` using one S-Function Builder Block. This block is selected to allow for the experimental implementation to use the same `C` code from simulations. The model of the machine was implemented in Simulink according to a stationary coordinate system ($\alpha\beta$) for simplicity and prioritizing the simulation speed.

The simulation time T_{sim} is an important parameter to adjust, it specifies the time in which all the equations of the machine are executed. All integrators of the model are configured to work in a continuous way. Normally, the T_{sim} value is calculated as $20T_s$, where T_s is the control sampling time. In order to control the machine, a two-level inverter connected to a direct voltage source $v_{dc} = 540$ (V) is considered. This topology generates eight different switching states and seven different voltage vectors. The inverter is modeled in Simulink by simplicity. In order to sample the inputs every T_s seconds, a ZOH block is considered. The general configuration for all simulations is shown in Fig. 4.8

In Fig. 4.8 it is possible to notice that a speed control loop is considered using a PI-controller, which generates the torque reference T^* for the inner control system. This controller is configured using a discrete-time model of the mechanical part of the induction machine. For the speed measurement, a different sampling time $T_{s\omega}$ is considered, this value is bigger than the sampling time T_s used by the control system in order to avoid measurement noise in the inner loop. For simulations, the speed is sampled using a down-sampled routine with $T_{s\omega} = 1$ (ms). Finally, parameters of simulations are shown in Table 4.2

4.5.2 Implemented PTC Algorithm

The objective of conventional PTC control scheme objective is the optimization (minimization) of the cost function presented in (4.27). The predictive algorithm evaluates, at every sampling time, all possible voltage vectors, and then selects the one that returns the minimum value for (4.27) to be applied in the next sampling instant. The control scheme is summarized in the flow diagram illustrated in Fig. 4.9, and it is completed in following steps:

- Step 1 *Measurement*: Sampling to get $\mathbf{i_s}^k$, v_{dc}^k and ω^k.

Table 4.2: Simulation parameters

Description	Parameter	Value
Induction Machine Drive		
Stator Resistance	R_s	1.6647 (Ω)
Rotor Resistance	R_r	1.2134 (Ω)
Stator Inductance	L_s	136.82 (mH)
Rotor Inductance	L_r	136.82 (mH)
Magnetizing Inductance	L_m	130.69 (mH)
Pair Poles	p	2
Total Inertia	J	0.02398 (kgm^2/s)
Nominal Speed	ω_{nom}	1440 (rpm)
Nominal Torque	T_{nom}	25 (Nm)
Nominal Power	P_{nom}	4.0 (kW)
dc-Link	v_{dc}	540 (V)
Predictive Controllers		
Simulation Time	T_{sim}	2 (μs)
Sampling Time	T_s	40 (μs)
Flux Reference	$\|\|\mathbf{\Psi_s}^*\|\|$	0.98 (Wb)
Magnetization Direct Current	i_{sd}^{mag}	7.115 (A)
Flux Weighting Factor	k_Ψ	4096.0
Torque Weighting Factor	k_T	1.0
Quadrature Weighting Factor	k_q	1.0
Direct Weighting Factor	k_d	0.61
Speed Controller		
Speed Sampling Time	$T_{s\omega}$	1 (ms)
Speed Reference	ω^*	1440 (rpm)
Saturation Quadrature Current	i_{sq}^{sat}	9.381 (A)
Load Torque	T_l	12.5 (Nm)
Discrete-Time Proportional Gain	$k_{p\omega}$	0.39562
Discrete-Time Integral Gain	$k_{i\omega}$	0.38691636
Controller Bandwidth	BW_ω	10 (Hz)

- Step 2 *Apply*: Set the optimal voltage vector $\mathbf{v_s}_{opt}^k$ found in the previous loop iteration.

- Step 3 *Estimate*: Flux estimations by using (4.16) and (4.19).

- Step 4 *Evaluate*: Extrapolate the control variables for $\mathbf{v_s}_{opt}^k$ using (4.21) and (4.22). Predict the control variables for every possible voltage vector $\mathbf{v_s}^{k+1}(j)$, with $j = 0, ..., 6$ using (4.23) and (4.24). Then, evaluate $G(\mathbf{v_s}^{k+1})$ using (4.27).

- Step 5 *Optimize*: Select optimal $\mathbf{v_s}_{opt}^{k+1}$ (minimization). Return to Step 1.

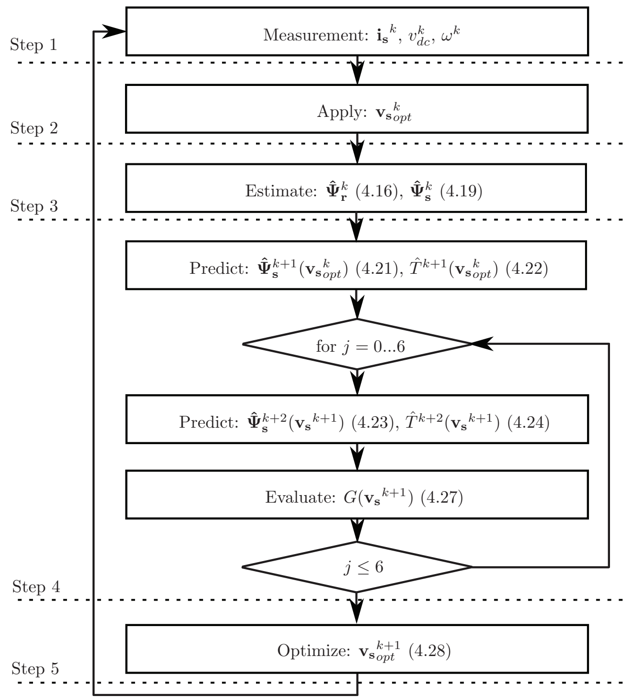


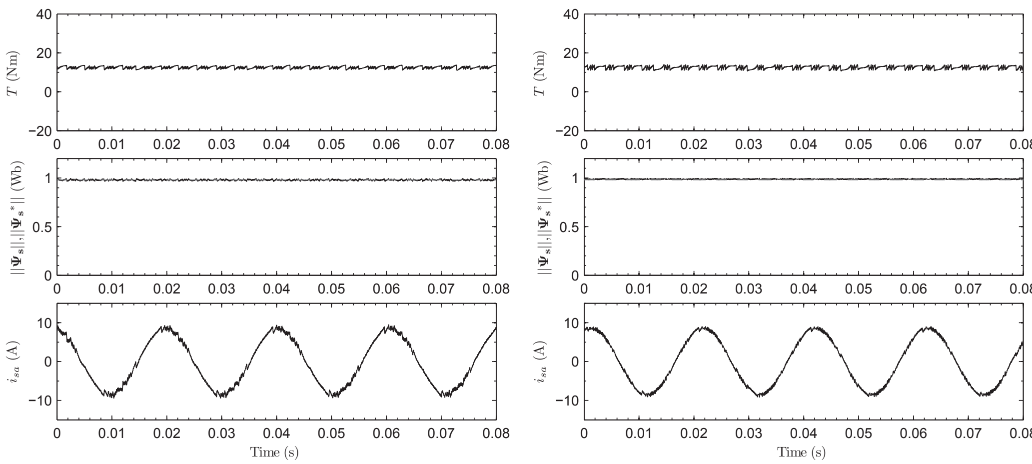
Figure 4.9: Predictive torque and flux control algorithm considering the calculation delay compensation.

4.5.3 Results

The implemented cost function is presented in (4.27) considering the switching scheme based on active vectors with zero-redundances illustrated in Fig. (4.5b) due to its simplicity and low stator current THD. The used approximate discretization model is the modified Taylor method because it corresponds to the best approximation to the continuous model according to the study presented in the previous chapter.

4.5.3.1 Steady-State Operation

The first simulation presents the steady state behavior for the PTC control strategy when the machine is operating at a nominal motoring speed at 1440 (rpm) with 50 (%) of the load torque, 12.5 (Nm). Fig. 4.10a shows the sinusoidal waveform of the stator current, the constant electric torque, and stator flux in steady state. The stator flux reference is the nominal one at 0.98 (Wb). The stator flux produces good current waveforms and, as a consequence, good torque performance in steady state. Fig. 4.10b presents the steady state behavior for the PCC control strategy. The stator current in PCC has low distortion with its THD slightly lower compared to PTC. Performance indices obtained with PTC and PCC, in steady state, are presented in Table 4.3.



(a) (b)

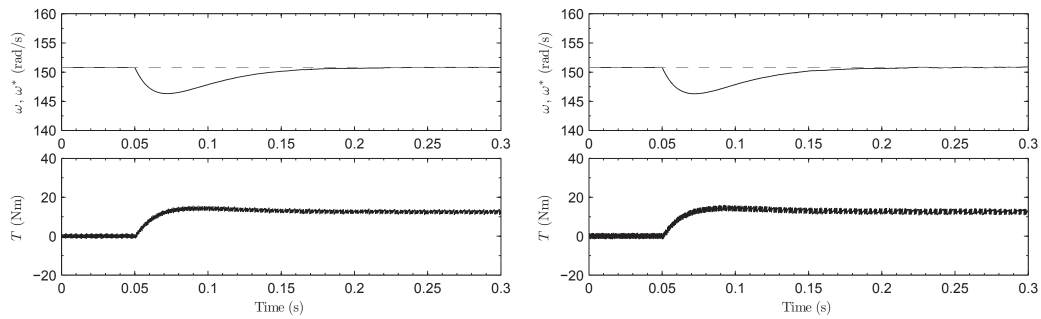
Figure 4.10: Simulation results using the: (a) PTC scheme; (b) PCC scheme, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

Table 4.3: Indices by using different discrete-time models for PTC and PCC

Indices	Forward Euler	Matrix Fact.	Taylor	Taylor Modified
Predictive Torque Control				
THD_{is} (%)	6.771	6.779	6.724	6.779
THD_{T} (%)	6.574	6.824	6.665	6.870
NRSMD_{Ψ_s} (%)	0.654	0.648	0.652	0.654
NRSMD_{T} (%)	4.700	4.887	4.789	5.040
$\mathrm{NRSMD}_{\hat{\mathbf{\Psi}}_\mathbf{r}}$ (%)	0.052	0.042	0.048	0.030
$\tilde{f}_{sw}$ (kHz)	3.013	3.011	3.010	3.010
Predictive Current Control				
THD_{is} (%)	5.113	5.079	5.114	4.967
THD_{T} (%)	9.121	8.999	9.121	8.907
NRSMD_{Ψ_s} (%)	1.445	1.214	1.450	0.971
NRSMD_{T} (%)	7.351	6.995	7.352	6.668
$\mathrm{NRSMD}_{\hat{\mathbf{\Psi}}_\mathbf{r}}$ (%)	0.037	0.033	0.037	0.036
$\tilde{f}_{sw}$ (kHz)	2.528	2.577	2.528	2.558

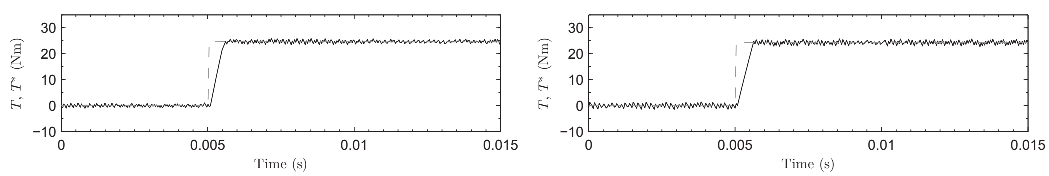
4.5.3.2 Load Impact

The second result shows the performance of PTC and PCC under a load torque impact of 12.5 (Nm) while the machine is running at the nominal speed. In Fig. 4.12, the behavior of the speed and torque is observed. Since the same PI-speed controller has been used, the



(a) (b)

Figure 4.11: Simulation results using the: (a) PTC scheme and (b) PCC scheme, in an induction machine fed by a 2L-VSI. Speed and torque behavior during a load impact of 50 (%) of the nominal load.



(a) (b)

Figure 4.12: Simulation results using the: (a) PTC scheme and (b) PCC scheme, in an induction machine fed by a 2L-VSI. Torque step response, (gray) torque reference, and (black) torque response.

rotor speed and torque reacts almost in the same way for both strategies.

4.5.3.3 Torque Response

The third simulation presents the dynamic behavior of the torque when the machine is operating at nominal flux condition. In Fig. 4.12, the torque step is equal to 25 (Nm), and it was performed by a change in the speed reference from 0.33(%) to 0.66(%) of nominal speed. A very quick transient response is obtained as with other direct strategies, due to the absence of an internal linear current control loop. As with standard PTC, the stator flux and torque are directly controlled by the optimal voltage vector selected in the previous iteration. The ripple of the torque is higher compared with a linear technique, such as FOC; however, the transient response is improved [34]. Fig. 4.12a and Fig. 4.12b present the torque response for the PTC and PCC control strategy, respectively. Both schemes have the same torque response; however, the torque ripple of PTC is lower compared to PCC, due to that in PTC torque is directly controlled.

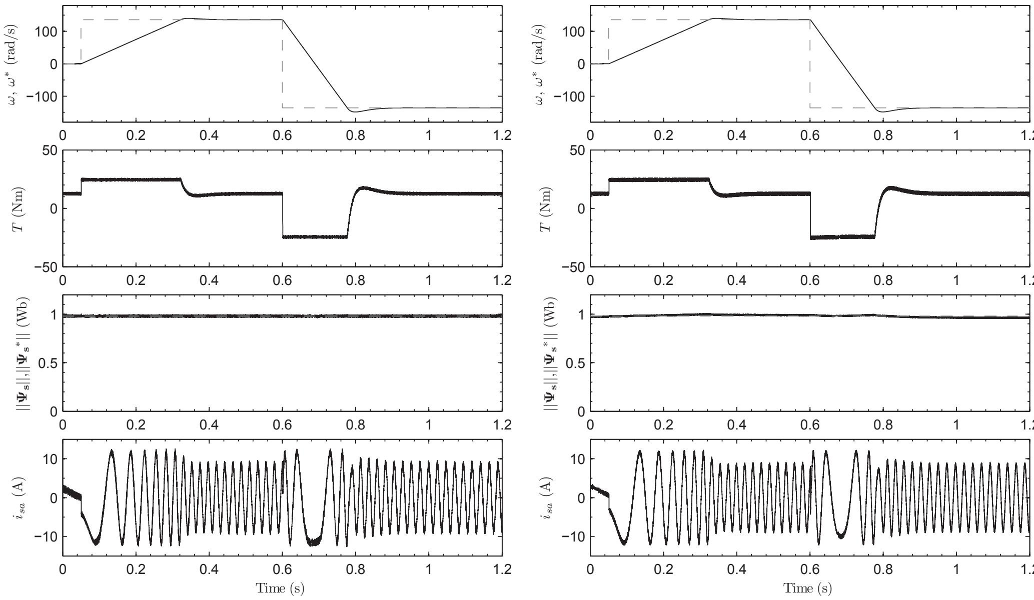


Figure 4.13: Simulation results using the: (a) PTC scheme and (b) PCC scheme, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

4.5.3.4 Speed Reversal Maneuver

Finally, the last simulation illustrates a speed-reversal operation from 0.9(%) to -0.9(%) of ω_{nom}. Fig. 4.13a shows the speed, torque, stator flux, and current waveform dynamics. The speed control is done with a PI controller, and for this reason, the speed has a smooth response. It is not the case of the torque and the stator flux due to the use of a nonlinear control law. Furthermore, a decoupled control of the electric torque and the stator flux is achieved with the proposed PTC method. Fig. 4.13b shows the speed, torque, stator flux, and current waveform dynamics using the PCC control strategy. Both schemes have similar dynamic responses, but the tracking of stator flux in PTC is better than the obtained with PCC. The above occurs because in PCC the control variable is $\mathbf{i_s}_{dq}$, while in PTC the control variables are stator flux and torque directly.

4.5.3.5 Prediction Error

Stator current prediction is almost the same for both models, while rotor flux direct and quadrature components differ, mainly due to the different structures of the $\mathbf{B_d}$ matrix for each discretization method [28]. Comparison was made using the distance to the continuous-time space vector shown in Fig. 4.14, with error as percentage of the maximum continuous time vector norm. The error of each model, has been tested for the maneuver shown in Fig. 4.14a. As expected in the case of PTC, the Taylor Modified has the lowest error during the maneuver, while for PCC the matrix factorization has the lowest error during the maneuver.

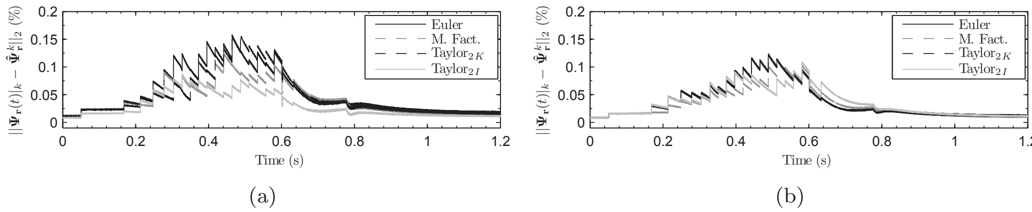


Figure 4.14: Rotor flux prediction error using different discrete-time models for: (a) PTC scheme and (b) PCC scheme.

Obtained error values for both strategies are presented in Table 4.3.

4.6 Conclusions

The specification of weighting factors is a very complex task in the implementation of predictive torque control. When more objectives are considered in the total cost function, the weighting factors calculation is usually performed using trial and error procedures and running time-consuming simulations. An alternative to predictive torque control has been reported. In fact, the alternative finite control set model predictive field-oriented control or PCC represents a simplification for weighting factor problem.

In terms of simulation results, the obtained THD of PCC is slower than PTC scheme. However, torque and stator flux are not directly controlled. Finally, although the use of the linear combined objective function to solve the optimization problem at each sampling time is relatively simple, for high performance a set of weighting factors must be calculated *a priori*.

Chapter 5

MULTIOBJECTIVE FCS-MPC

5.1 Introduction

ENABLED by the actual computational power available, new control techniques that previously were very difficult to implement are increasingly being considered for power converter and drives control. Controlling this kind of systems is a complex task due to their inherent hybrid or switched nature [53]. This characteristic appears because these systems are composed by a part with continuous states and another with discrete ones, i.e., the switching matrix, implemented through on-off switches. The desired behavior, reflected on the continuous part, is obtained changing switching matrix states through a proper control strategy. The conventional approach to control power converters is to discard the switched characteristic of the system through modulation [53]. However, considering power converters as hybrid systems is an attractive approach because it could allow higher dynamic performance. Switched systems have received increasing attention in the control community in past decades, and some applications have been reported in power electronics.

From the control techniques that consider power converters as hybrid systems, Finite Control Set Model Predictive Control (FCS-MPC), has been rising as a promising control technique [11]. In this control method the discrete-time system model is evaluated for every possible converter actuation and then compared with the signal reference in order to select the best voltage vector [32]. Due to its basic concept, it is restricted to deterministic power conversion systems for which a model can be derived through means of standard circuit laws. As a nonlinear control strategy, it could consider linear and non-linear system models and constraints. Two important technological limitations are that sampling and commutation frequencies limitation (i.e., approximately less than 50 (kHz)), given the computational power required to evaluate the system states and the response of semiconductors, respectively. On the other hand, it features fast dynamic response and flexibility [32]. FCS-MPC's potential has been shown by its application to different topologies [31].

FCS-MPC is based on cost functions that represent desired goals, e.g., current reference tracking, capacitor balancing, active and reactive power control, switching frequency, efficiency, common-mode voltage. Then, when one control objective is desired, only one cost function must be minimized. However, many applications have more than one control objective. To solve this, an aggregated objective function (AOF) is constructed as a linear combination of individual cost functions using so-called weighting factors. Every sampling time, a switching state is selected to be applied at the following one. The selected state minimizes the AOF two sampling times ahead, in the set of all valid switching states [32,59].

The main issue of using weighting factors is their correct selection, because it is more complex than tuning PI parameters or hysteresis bands on conventional controllers [41,43]. Controller performance is directly related to weighting factor selection [43]. Their main function is to try to model the relative importance among desired control objectives [45]. In contrast to the tuning of conventional controllers, FCS-MPC currently lacks tools to systematically select its parameters. Although several methods to find appropriate weighting factors have been reported, they are based on empirical [42,44,45] or offline procedures [43], and are time consuming and not systematic as reported in the previous chapter. In addition, they depend on the system operating point and its parameters [43]. As tuning weighting factors is a key issue in FCS-MPC and an avoidance of them could be an interesting option. This drawback could be mitigated replacing the weighting factor based stage with a multiobjective (MO) formulation [8,50].

5.2 Multiobjective Optimizations

A topic of growing interest in engineering and economics is the solution of multiobjective optimization problems. Solving them involves the use of both optimization and decision methods, since their final solution consists of the optimal point that best fits the interests of someone who deeply knows the problem, the decision maker. The multicriteria analysis studies manners of aiding man to make decisions of uncertainty or conflicting interest. Its main goal is to maximize the coherence between the final decision and the decision maker interests, taking into account the capacities of designer rationality. There are two main lines of thought concerning multicriteria analysis: the American and the French schools [75].

The first one is based on the multi-attribute utility theory, which states that all objectives can be combined into a cost function, which assigns a number to each available alternative. By ordering these numbers, it is possible to order all alternatives according to the decision maker preferences. The favorite solution is the one associated to the highest number.

The second one is based on the outranking concept. The outranking relations are defined between every pair (a, b) of alternatives, in such manner that if a is better than b (according to the decision maker interests), then it is said that a outranks b. Having built these relations, they are exploited according to some rules in order to obtain the most satisfactory solution.

There are different manners of choosing the most suitable efficient solution of multiobjective optimization problems, such as *a priori decision*, *a posteriori decision* and *non-scalarization methods*. In *a priori* decision method there is a priori articulation of preferences, where the original multiobjective problem is transformed in a single objective problem through weighting factors. In *a posteriori* decision method, first a multiobjective search is executed and, after that, a decision method or decision stage is applied to the obtained efficient front [49].

The most well-know multiobjective optimization in MPC is the *a priori* decision method based on an aggregated objective function (AOF). While, for *a posteriori* decision two methods are studied in this chapter: Ranking Method, based on outranking relations and and fuzzy decision algorithm based on the Bellman-Zadeh approach [52]. A general classification of multiobjective optimization methods are presented in Fig. 5.1 [49].

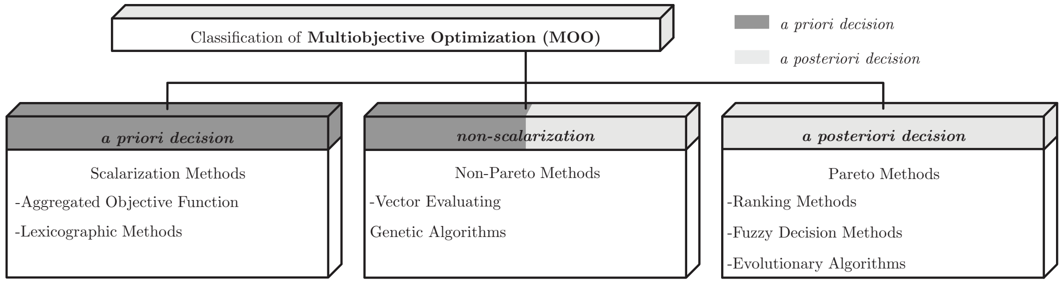


Figure 5.1: Classification of predictive control methods used in power converters.

In lexicographic aggregation the objectives are optimized in their order of the importance while in fitness combination parameters of the aggregating function reflect the human preferences. When the preference involved could be faithfully captured in the mathematical model employed and no practical computational difficulties arise, *a priori* approach results simple and efficient. However, this case need a high knowledge of the control problem, e.g., PTC based on AOF. Some objectives often cannot be adequately modeled in a priori preference specification. Further, a priori approaches often require sufficient knowledge of the specific problem before associated parameters in the aggregation function could be determined [76].

5.2.1 Multicriteria Decision-Making

Over more than 40 years, many literature surveys and bibliographies have been published in the area of Multicriteria Decision-Making (MCDM) and the development of subfields these were mostly devoted to particular aspects of multicriteria or multiobjective optimization, e.g., Multiobjective Integer Programming, Multiobjective Combinatorial Optimization, Vector Optimization, Multiobjective Evolutionary Methods, Fuzzy Multiobjective Programming, Applications of MCDM, Goal Programming and others. Some of these methods are incorporated to the MPC formulation, giving rise to Multiobjective Optimal Control [49]. The multiobjective optimization problems can be classified by general in two cases. The first corresponds to the Multiobjective Optimization (MO), where the solutions are infinite. Furthermore, a mechanism exist to generate candidate solutions. On the other hand, when the solutions are finite and known, the problem is called MCDM.

5.2.2 Optimal Solutions in MCDM

The MCDM problem try to find the best solution $\mathbf{u}_{opt}$ (optimal solution), considering multiple cost functions (n). This cost function vector is named

$$\mathbf{G}(\mathbf{u_j}) = [g_1 \ g_2 \ \cdots \ g_n]^T, \tag{5.1}$$

with n the total number of cost functions. The cost function vector is evaluated for all the possible alternatives and the optimization problem is derived as

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}_j \in U} \mathbf{G}(\mathbf{u}_j), \tag{5.2}$$

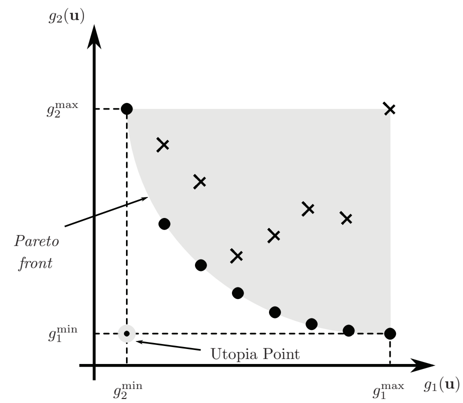


Figure 5.2: Pareto optimal solutions considering a problem with two objectives.

where $\mathbf{u}_j$, $j = 1, \cdots, r$, corresponds to the j-th alternative to consider in the set U and r the total alternative number, i.e., in a 2L-VSI induction motor drive, there are eight alternatives to evaluate (voltage vectors, $j = 8$) and two cost functions (torque and flux control). The major issue in this kind of optimization is that generally there is no alternative that minimizes all objective functions simultaneously. The selection of the optimal solution implies a balance or compensation (*trade-off*) between different evaluation criteria.

5.2.2.1 Pareto Front

When the objective functions are said to be conflicting, and there exists infinite number of optimal solutions named nondominated or Pareto Solutions. A solution is called *Pareto optimal*, if there does not exist another solution that dominates it. The set of Pareto optimal outcomes is often called the *Pareto front* [40].

$$U_{\text{Pareto}} := \{\mathbf{u}_a \in U : \nexists \mathbf{u}_b \in U : \forall i, g_i(\mathbf{u}_b) \leq g_i(\mathbf{u}_a) \wedge \exists k : g_k(\mathbf{u}_b) < g_k(\mathbf{u}_a)\}, \tag{5.3}$$

where $a, b = \{1, \cdots, r\}$, $a \neq b$. These solutions appear over the *Pareto front*, e.g., considering a problem with two objectives, the *Pareto front* is illustrated in Fig. 5.2. Nondominated and dominated solutions are defined by the alternatives represented with a solid point and cross, respectively. Finally, considering the *Pareto front* there are not a finite set of nondominated solutions, the best solution is determined by the decision maker method, where one alternative is to minimize the distance between the *Utopia Point* and the nondominated solution.

5.3 Multiobjective Methods in FCS-MPC

In the traditional formulation of the FCS-MPC scheme, the controller tries to minimize the cost functions for each particular objective, minimizing an aggregate objective function (AOF) composed by a linear combination of them. In this section two basic multiobjective optimization strategies are proposed to replace the AOF by a multiobjective optimization stage allowing a fair optimization of the required control objectives. The first method is based in a technique applied to the ranking of populations in evolutive optimization algorithms based on genetic algorithms [46, 51], but it is simplified significantly since the possible solutions is a finite control set. The second method is based on the well-known fuzzy multicriteria decision-making (FMCDM) or fuzzy decision-making (FDM) [47, 52].

5.3.1 Aggregate Objective Function (AOF)

This is probably the most widely used MOO method. It consists in assigning a non-negative weight ($k_i > 0$) to each the i-th objective function, so that the overarching scalar objective function can be expressed as

$$G(\mathbf{u}_j) = \sum_{i=1}^{n} k_i g_i(\mathbf{u}_j) = \mathbf{k}^T \mathbf{G}(\mathbf{u}_j). \tag{5.4}$$

where $\mathbf{k} = [k_1\ k_2\ \cdots\ k_n]^T$ is the weight vector and $\mathbf{G}(\mathbf{u}_j) = [g_1\ g_2\ \cdots\ g_n]^T$ is the cost function vector [50]. Remember that, each objective function has an associated error value at each sampling time, but in the following analysis is omitted by nomenclature simplicity. Then, the scalar cost function is evaluated for all the possible converter actuation and the optimal solution is given by

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}_j \in U} G(\mathbf{u}_j). \tag{5.5}$$

Commonly, weight vector are chosen as $||\mathbf{k}||_1 = 1$. Moreover, the objective functions have to be normalized since not all objectives have the same range of values. The most important advantage of this method is the transformation of the vector objective function ($\mathbf{G}$) in a single-objective function (G), such that traditional optimization methods can be used [50]. The problem is the setting of the weights and the results are sensitive to weights ratio and they are difficult to be chosen. On the other hand, weights indicate the relative importance of the corresponding objective function but they do not mean priorities [50]. Hence, if the optimization process cannot be completed for all objective functions, the method does not indicate, in which sequence objective functions may be discarded. Moreover, the method presents difficulties in case of non-convex problems due to nondominated solutions are not completely modeled [50].

5.3.2 Multiobjective Ranking-Based (MRB)

In the above method, there is an *a priori decision*, where the original multiobjective problem is transformed in a single objective problem. It is natural to think that a variation in the optimization method used above, e.g., using an *a priori decision* or progressive articulation of preferences could allow for the elimination of the weighting factors, and therefore the elimination of its selection, replacing them by a decision algorithm (decision stage). In general, the main requirement is to minimize the desired cost function of each objective, while a second request is to achieve a fair minimization in all goals. The main characteristic of studied methods is that the controlled variables have the same priorities in the control scheme.

Another issue in the AOF method is that the tuning of weighting factors is a complex task, particularly when a greater number of goals are desired. For this reason, it would be interesting to avoid the selection process of the weighting factors. Here, a variation of the standard optimization method is presented making the tuning of weighting factors unnecessary. The method makes use of a multiobjective algorithm to solve the optimization problem, considering a fair selection of the solution with respect to the control objectives.

The Multiobjective Ranking-Based method (MRB) is based in a technique applied to the ranking of populations in evolutive optimization algorithms based on genetic algorithms [51], but it is simplified significantly since the possible solutions are finite. The strategy is to evaluate each objective function separately for each possible solution. After obtaining these values, the error evaluation of each objective function are sorted according to a ranking, which assigns a higher position (worst) to higher values, while states with lower values are assigned a lower ranking (better). That is,

$$g_i(\mathbf{u}_j) \longrightarrow r_i(\mathbf{u}_j), \tag{5.6}$$

where g_i is the i-th cost function, $\mathbf{u}_j$ is the j-th valid converter actuation and the ranking associated to the i-th objective function. In fact, for each cost function (objective) there is a ranking that determines the relative quality of each possible solution (e.g., voltage vector) with respect to all the other solutions. The described procedure allows obtaining the best solutions for each objective.

The main idea is the independent evaluation of each objective function for the j-th valid converter actuation, and then calculating a ranking of each possible solution using a sorting algorithm. To select which is the best overall solution with respect to all possible converter actuation, an overall criterion is needed. Here, are presented three different cases.

5.3.2.1 Average Ranking

To select which is the best overall solution an average scheme can be applied where the converter actuation with lower average ranking is selected, which can be expressed as

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}_j \in U} \frac{1}{n} \sum_{i=1}^{n} r_i(\mathbf{u}_j), \tag{5.7}$$

5.3.2.2 Distance Criterion (Euclidean Norm)

A variation of (5.7) consists of considering a distance criterion in which the converter actuation with lowest Euclidean norm of its rankings to the origin is selected,

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}_j \in U} \sqrt{\sum_{i=1}^{n} \left(r_i(\mathbf{u}_j)\right)^2}, \tag{5.8}$$

which can be interpreted as the solution with the lowest distance to the best possible point for both objectives.

5.3.2.3 Pareto Filter

In the case of single objective optimization it is relatively simple to determine if a solution is better than other. When the problem has a higher number of objectives the situation is more complex because a solution can be better in a certain objective but worse in other. To consider this problem the concept of Pareto optimal solution can be used, which is presented in (5.3). The nondominated solutions can be considered as the best compromised solutions for the problem and they conform the Pareto front. According to different priorities, it is possible to select the best solution for a particular problem. This concept allows the proposal of another method of selection of the best converter actuation in which after filtering the nondominated solutions, a decision algorithm selects an optimum solution in a determined sense.

In the case of two control objectives, the decision algorithm sorts according to a ranking the nondominated solutions for each objective function. Later, the selected state is the one that has the least difference between both rankings,

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}_j \in U_{\text{Pareto}}} |\tilde{r}_1(\mathbf{u}_j) - \tilde{r}_2(\mathbf{u}_j)|, \tag{5.9}$$

where U_{Pareto} is the set of states obtained after the filtering, and $\tilde{r}_1$ and $\tilde{r}_2$ are the rankings of the j-th nondominated state. This strategy tries to ensure that the control for each variable is done in a fair way. In this case, it is not possible to use the average or distance rank because all the obtained solutions are optimum in this sense, and for this reason they have equal averages, in addition to the fact that among them, relatively unfair solutions between different objectives can be found. Average and Euclidean norm are relatively simple compared with Pareto filter strategy. In fact Pareto filter adds considerable computational burden.

5.3.3 Fuzzy Decision-Making (FDM)

To select an optimal solution there are two ways to proceed. The first is obtaining the Pareto front, the set of all nondominated solutions, and then selecting one from this set through a decision strategy. The other way is to directly select a solution, without first determining the Pareto front. Due to the computational complexity involved, the second option seems more appropriate to realtime applications with fast sampling times, such as a power converter. The proposed algorithm in section reflects this type of selection.

A decision-making strategy that has been developed and used with good results in various fields (e.g., Economics and Finance Applications [77] and Power System Applications [78,79]) is Fuzzy Multicriteria Decision-Making (FMCDM) [52]. In particular, it has been applied extensively to the solution of multiobjective optimization problems, where multiple and conflicting objectives must be met. FMCDM is characterized by the use of membership functions. These functions represent the degree of attainment of the goals for each solution. Then, a decision is obtained as the intersection or confluence of them.

The application of FMCDM to MPC has been introduced in [80,81]. This application of fuzzy logic is a bit different from the traditional approach used in control applications. The conventional approach derives a control action from the actual error and its change through a set of rules. In contrast, in FCS-MPC the specification of the preferences of the actuation selector is done from a set of given goals.

To apply FMCDM in FCS-MPC, the form of the membership functions and the type of confluence should be specified. A commonly used form of membership function is the linear one, as shown in Fig. 5.3a. it is defined as

$$\mu_i(\mathbf{u}_j) = \frac{g_i^{\max} - g_i(\mathbf{u}_j)}{g_i^{\max} - g_i^{\min}}, \tag{5.10}$$

where μ_i is the i-th goal membership function, $g_i^{\max}$ and $g_i^{\min}$ are defined as,

$$g_i^{\max} = \max\{g_i(\mathbf{u}_1), \cdots, g_i(\mathbf{u}_r)\}, \tag{5.11}$$

$$g_i^{\min} = \min\{g_i(\mathbf{u}_1), \cdots, g_i(\mathbf{u}_r)\}. \tag{5.12}$$

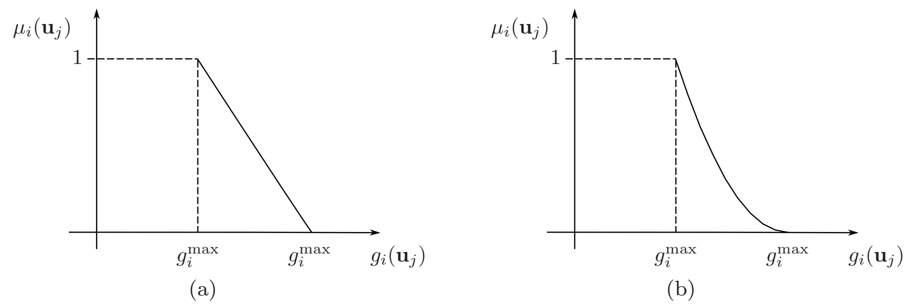


Figure 5.3: Membership functions used in FMCDM: (a) linear and (b) quadratic.

An alternative to (5.10) is the quadratic membership function presented in (5.13) and illustrated in Fig. 5.3b. This function is a good alternative when there is a strong interaction between objectives.

$$\mu_i(\mathbf{u}_j) = \left(\frac{g_i^{\max} - g_i(\mathbf{u}_j)}{g_i^{\max} - g_i^{\min}}\right)^2, \tag{5.13}$$

Each membership function constitutes a mapping from the space where the cost functions vary to the range [0,1]. Then, the resulting values are compatible and could be directly compared. The form of the membership functions could be understood as the type of scale used for this mapping and comparison. In general the each membership function can be defined with a priority weight k_i,

$$\mu_i(\mathbf{u}_j) = \left(\frac{g_i^{\max} - g_i(\mathbf{u}_j)}{g_i^{\max} - g_i^{\min}}\right)^{k_i}, \tag{5.14}$$

where k_i is the priority weight of the i-th criterion. Commonly, As in AOF, weight vector are chosen as $||\mathbf{k}||_1 = 1$. In particular, the linear mapping with same priority for each control objective has the benefit of low computational requirements.

From the membership function definition, it can be seen that the mapping varies at each sampling time. In this way, the scale used for each cost function is variable. Therefore, the possible levels of achievement of each goal are considered. These levels can vary significantly each sampling time, depending on the system characteristics, operation point and the cost functions used. Then, it is possible to make a direct comparison among the achievable optimization levels for each objective.

The final selection of the best converter actuation is performed by the decision function. For its specification, the designer of the strategy sets the preference for each goal. Considering the goals as equally important, three types of functions could be used. These are the maximizing decision, minimizing decision and the Hamacher logical `AND`.

5.3.3.1 Bellman-Zadeh Maximizing Decision

The Bellman-Zadeh maximizing decision, maximizing decision or `MIN` Operator is given by the maximum of the intersection membership function defined by

$$\mu_D(\mathbf{u}_j) = \min\{\mu_i(\mathbf{u}_1), \cdots, \mu_n(\mathbf{u}_r)\}. \tag{5.15}$$

Then, the best converter actuation to be applied at the next sampling time is selected as the one with the maximum value of the confluence function $\mu_D(\mathbf{u}_j)$. Thus,

$$\mathbf{u}_{opt} = \arg \max_{\mathbf{u}_j \in U} \mu_D(\mathbf{u}_j). \tag{5.16}$$

5.3.3.2 Bellman-Zadeh Minimizing Decision

Another well-known confluence function is the `MAX` Operator. Now, if the membership functions are defined as

$$\mu_i(\mathbf{u}_j) = \frac{g_i(\mathbf{u}_j) - g_i^{\min}}{g_i^{\max} - g_i^{\min}}, \tag{5.17}$$

the minimizing decision is given by the minimum of the intersection membership function defined by

$$\mu_D(\mathbf{u}_j) = \max\{\mu_i(\mathbf{u}_1), \cdots, \mu_n(\mathbf{u}_r)\}. \tag{5.18}$$

Then, the best converter actuation to be applied at the next sampling time is derived,

$$\mathbf{u}_{opt} = \arg \min_{\mathbf{u}_j \in U} \mu_D(\mathbf{u}_j). \tag{5.19}$$

5.3.3.3 Hamacher logical `AND` (Product Operator)

On the other hand, the Hamacher logical `AND` is defined by

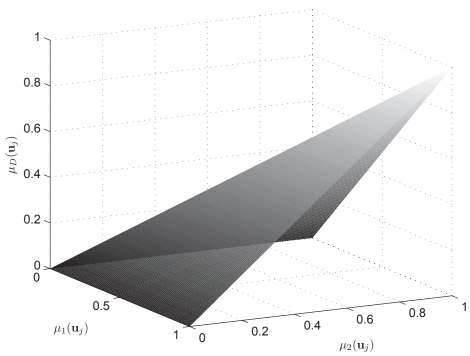
$$\mu_D(\mathbf{u}_j) = \prod_{i=1}^{n} \mu_i(\mathbf{u}_j). \tag{5.20}$$

where $\mu_i(\mathbf{u}_j)$ is defined in (5.10). This operator considers some degree of interaction among the objectives. In contrast, the maximizing decision always maximizes the fulfillment of the objective with the poorest achievement. It could bring better performance for a higher number of goals. In general, the choice of operator is application dependent [80]. More information about these operators can be found in [81].

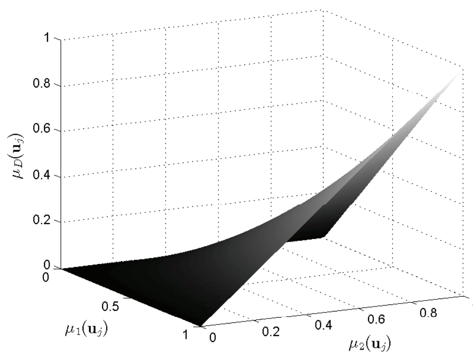
In the case of two control objectives and using linear membership functions, the representation of a decision function based on the `MIN` operator and `AND` operator are illustrated in Fig. 5.4a and Fig. 5.4b, respectively,

5.4 Conclusions

Conventional and two new multiobjective optimizations are presented in this chapter. Although the use of the AOF function to solve the optimization problem at each sampling time is simple, where the *a priori* specification of weighting factors is a very complex task in the implementation of predictive schemes. In the first proposed method, the weighting



(a)



(b)

Figure 5.4: Decision functions used in FMCDM: (a) `MIN` operator and (b) `AND` operator.

factor tuning is replaced by a multiobjective ranking-based approach, and weighting factor calculation is avoided. The method is based on the idea that the selected voltage vector should allow a fair minimization of all the objective functions. The second alternative is based on FMCDM, it allows the design of the FCS-MPC actuation selector from a higher level approach, instead of tuning weighting factors as in the standard scheme. The discussion has been limited to an algorithm which selects voltage vectors that optimize the required control objectives to the same degree at each sampling time. Naturally, this imposes a fixed tradeoff to the selection stage.

Chapter 6

MULTIOBJECTIVE PTC

6.1 Introduction

TWO basic multiobjective optimization strategies are proposed to replace the aggregate cost function allowing a fair optimization of the required control objectives. The first method is based in a technique applied to the ranking of populations in evolutive optimization algorithms based on genetic algorithms [51, 76]. The second method is based on the well-known fuzzy decision-making (FDM) [52] to avoid the weighting factors selection. Some multiobjective approaches has already been reported in Economics and Finance Applications [77], Power System Applications [78, 79] and recently in Model Predictive Control (MPC) [50,80,81], but not in the context of avoiding weighting factors in FCS-MPC of power converters. The main contribution of this chapter is to illustrate the feasibility of above approaches in FCS-MPC using simulations.

6.2 Aggregate Objective Function PTC

In the standard PTC scheme, the optimal voltage vector to apply in the next sampling time is selected, minimizing a simple cost function G. To obtain high performance, the weighting factors of this cost function should be selected [46]. Examining how the voltage vector is selected, the minimization of the single cost function can be recognized as a particular form of a multiobjective optimization called aggregate objective function [40].

The method considers the minimization of a single objective function that pursues torque and flux tracking at every sampling time T_s. Then, considering the elimination of the calculation delay in digital implementation, the cost function to minimize has the following structure:

$$G(\mathbf{v_s}^{k+1}) = k_T g_1(\mathbf{v_s}^{k+1}) + k_\Psi g_2(\mathbf{v_s}^{k+1}), \tag{6.1}$$

$$g_1(\mathbf{v_s}^{k+1}) = \left(T^* - \hat{T}^{k+2}\right)^2 \tag{6.2}$$

$$g_2(\mathbf{v_s}^{k+1}) = \left(||\mathbf{\Psi_s}^*|| - ||\hat{\mathbf{\Psi}}_\mathbf{s}^{k+2}||\right)^2 \tag{6.3}$$

where, the variables $g_1(\mathbf{v_s}^{k+1})$ and $g_2(\mathbf{v_s}^{k+1})$ are calculated using (4.23) and (4.24). Note that, they depend on the voltage vector ${\mathbf{v_s}}^{k+1}$. The torque reference T^* is externally generated by a PI-speed controller and $||{\mathbf{\Psi_s}}^*||$ is the stator flux reference. Thus, the stator voltage vector that minimizes (6.1) is selected to apply at the next sampling time,

$$\mathbf{v_s}_{opt} = \arg \min_{\{\mathbf{v_0},\ldots,\mathbf{v_7}\}} G({\mathbf{v_s}}^{k+1}), \tag{6.4}$$

where, $\mathbf{v_s}_{opt}$ is the optimal stator voltage vector to apply in the next sampling time. Remember that the weighting factors are selected as $k_T = 1$ and $k_\Psi = 4096$ according to a parameter sweep presented in the Chapter 4.

This method is widely used in conventional FCS-MPC approaches. The most important advantage of AOF is the transformation of the vector objective function ($\mathbf{G}$) in a single-objective function (G). The problem is the setting of the weights. The tuning of weighting factors is a complex task, particularly when a greater number of goals are desired. For this reason, it would be interesting to avoid the selection process of the weighting factors. A continuation, a variation of the standard optimization method is presented making the tuning of weighting factors unnecessary. The method makes use of a multiobjective algorithm to solve the voltage vector selection problem in PTC, considering a fair selection of the voltage vector with respect to the control objectives [46].

6.3 Multiobjective Ranking-Based PTC (MPTC)

The multiobjective optimization problem to solve at each sampling time in PTC can be stated as the fair minimization of the two different cost functions

$$g_1({\mathbf{v_s}}^{k+1}) = \left(T^* - \hat{T}^{k+2}\right)^2, \tag{6.5}$$

$$g_2({\mathbf{v_s}}^{k+1}) = \left(||{\mathbf{\Psi_s}}^*|| - ||\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}||\right)^2, \tag{6.6}$$

where $g_1({\mathbf{v_s}}^{k+1})$ and $g_2({\mathbf{v_s}}^{k+1})$ are the errors associated with the torque and stator flux, respectively. This simultaneous optimization can be interpreted as a vector optimum problem [40]. The proposed multiobjective ranking-based strategy evaluates these components separately for each possible voltage vector of the converter.

The operation of the Multiobjective Ranking-based PTC strategy (MPTC) is as follows. First, the values obtained from the evaluation of each objective function, $g_1({\mathbf{v_s}}^{k+1})$ and $g_2({\mathbf{v_s}}^{k+1})$, are sorted. Then, a ranking value is assigned to each error value. Voltage vectors with lower error are assigned a lower ranking, while voltage vectors with higher error are assigned a higher ranking. That is

$$g_1\left({\mathbf{v_s}}^{k+1}\right) \longrightarrow r_1\left({\mathbf{v_s}}^{k+1}\right), \tag{6.7}$$

$$g_2\left({\mathbf{v_s}}^{k+1}\right) \longrightarrow r_2\left({\mathbf{v_s}}^{k+1}\right), \tag{6.8}$$

where ${\mathbf{v_s}}^{k+1}$ are the evaluated voltage vectors and $r_1\left({\mathbf{v_s}}^{k+1}\right)$ and $r_2\left({\mathbf{v_s}}^{k+1}\right)$ are the ranking values associated with $g_1({\mathbf{v_s}}^{k+1})$ and $g_2({\mathbf{v_s}}^{k+1})$, respectively. The ranking value determines the relative quality of each possible voltage vector with respect to all the remaining possibilities. The associated ranking value is a dimensionless variable, while the error has a specific dimension (i.e., Nm and Wb). By selecting the ranking with the lowest value, it is possible to select an optimal voltage vector, from the point of view of one variable error (e.g., torque error). On the other hand, by selecting the ranking with the lowest value, it is possible to select an optimal voltage vector, from the flux error point of view.

Table 6.1: Example of voltage vector selection using MPTC

$\mathbf{v_s}$	$g_1(\mathbf{v_s})$	$g_2(\mathbf{v_s})$	$r_1(\mathbf{v_s})$	$r_2(\mathbf{v_s})$	$0.5(r_1+r_2)$	$\sqrt{r_1^2+r_2^2}$
$\mathbf{v_0}$	0.10	0.0013	0	3	1.5	3.00
$\mathbf{v_1}$	0.60	0.0012	5	2	3.5	5.38
$\mathbf{v_2}$	0.33	0.0002	3	1	2.0	3.16
$\mathbf{v_3}$	0.31	0.0001	2	0	$\boxed{1.0}$	$\boxed{2.00}$
$\mathbf{v_4}$	0.36	0.0027	4	6	5.0	7.21
$\mathbf{v_5}$	0.27	0.0025	1	5	3.0	5.10
$\mathbf{v_6}$	0.66	0.0015	6	4	5.0	7.21

6.3.1 Overall Criteria

Now, to select which one is the best overall voltage vector within these alternatives, an average criterion is used in which the voltage vector with the minimum average value of its rankings is selected, resulting in an equal compromise of tracking for both variables, torque, and flux. Then, the proposed optimization based on *average ranking* is

$$\mathbf{v_s}_{opt} = \arg \min_{\{\mathbf{v_0},\ldots,\mathbf{v_7}\}} \frac{r_1\left(\mathbf{v_s}^{k+1}\right) + r_2\left(\mathbf{v_s}^{k+1}\right)}{2}. \tag{6.9}$$

The optimization presented in (6.9) can be simplified as the sum of $r_1\left(\mathbf{v_s}^{k+1}\right)$ and $r_2\left(\mathbf{v_s}^{k+1}\right)$ only, because the optimization problem is equivalent. However, the average criterion was used to introduce the average ranking concept. This approach is commonly used in evolutionary optimization algorithms based on genetic algorithms [51, 82].

A variation of the *average ranking* approach consists of considering a distance criterion in which the converter actuation with lowest Euclidean norm of its ranking to the origin is selected,

$$\mathbf{v_s}_{opt} = \arg \min_{\{\mathbf{v_0},\ldots,\mathbf{v_7}\}} \sqrt{r_1\left(\mathbf{v_s}^{k+1}\right)^2 + r_2\left(\mathbf{v_s}^{k+1}\right)^2}, \tag{6.10}$$

which can be interpreted as the solution with the lowest distance to the best possible point for both objectives.

6.3.2 Calculation Example

To illustrate the proposed algorithm, a two-level inverter with eight voltage vectors is considered, where $\mathbf{v_s} = [\mathbf{v_0}\ \mathbf{v_1}\ \mathbf{v_2}\ \mathbf{v_3}\ \mathbf{v_4}\ \mathbf{v_5}\ \mathbf{v_6}\ \mathbf{v_7}]$, with $\mathbf{v_0}$ and $\mathbf{v_7}$ being two zero voltage vectors and $\mathbf{v_1}$ to $\mathbf{v_6}$ being the active voltage vectors. The utilization of the different zero vectors is alternating between $\mathbf{v_0}$ and $\mathbf{v_7}$, and for this reason, only one zero vector is used in the calculation example presented in Table 6.1. The torque and flux errors associated with the different voltage vectors are $g_1(\mathbf{v_s})$ and $g_2(\mathbf{v_s})$, respectively. Then, a ranking assignation for each error value is needed. For example, considering that the torque error associated with the voltage $\mathbf{v_0}$ is $g_1(\mathbf{v_0}) = 0.1$, the ranking value assigned is $r_1(\mathbf{v_0}) = 0$ because it is the lower error value, while the torque error associated with the voltage $\mathbf{v_6}$ is $g_1(\mathbf{v_6}) = 0.66$,

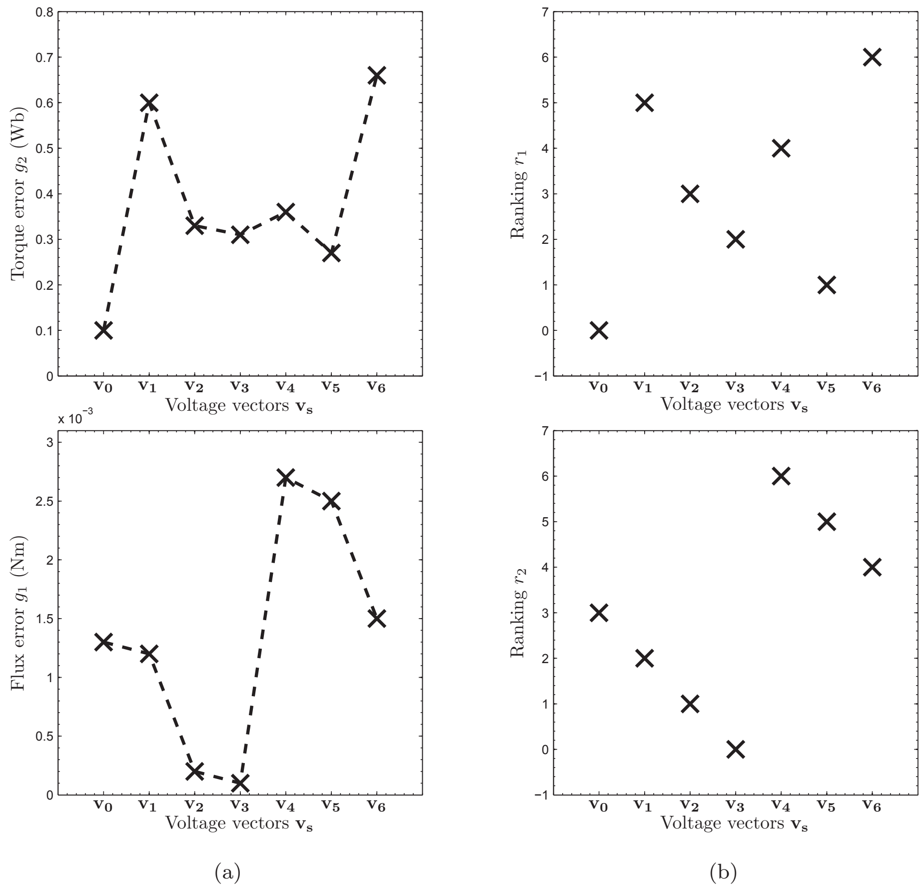


Figure 6.1: Graphical representation of Table 6.1: (a) errors and its (b) associated rankings.

and then, the ranking assigned is $r_1(\mathbf{v_6}) = 6$ because it is the higher error value. The same procedure should be performed for the flux errors with its respective ranking assignation r_2. Now, the average ranking value is calculated for each voltage vector $\mathbf{v_s}$, resulting in that the voltage vector $\mathbf{v_3}$ gives the lower average value of the rankings $r_1 = 2$ and $r_2 = 0$. Finally, the voltage vector to apply in the next sampling time is $\mathbf{v_3}$.

The calculation example presented in Table 6.1 is graphically explained in Fig. 6.1 and Fig. 6.2. Numerical errors and associated rankings are plotted in Fig. 6.1. Now, the average ranking value is calculated for each voltage vector $\mathbf{v_s}$, resulting in that the voltage vector $\mathbf{v_3}$ gives the lower average value of the rankings $r_1 = 2$ and $r_2 = 0$. This point is marked with with a circle in Fig. 6.2a. The same procedure is performed using an Euclidean-norm, where the same actuation voltage is selected as optimal. This point is marked with with a circle in Fig. 6.2b.

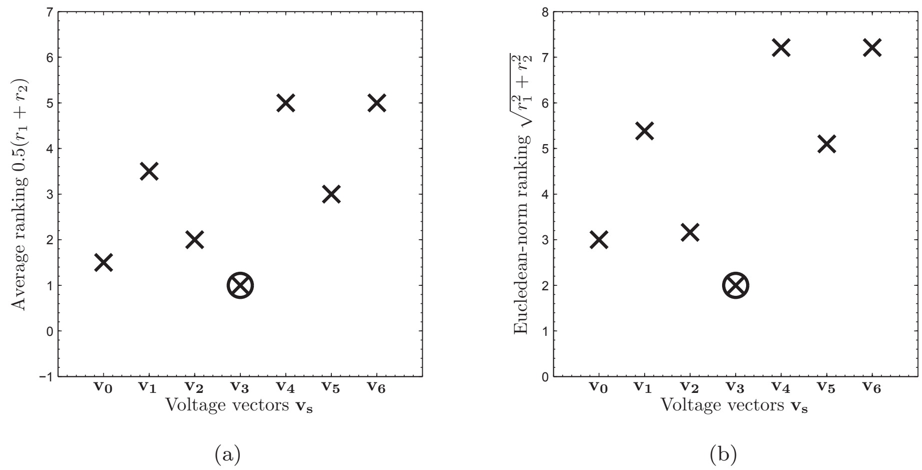


Figure 6.2: Example of Voltage Vector Selection 6.1: (a) average and (b) Euclidean-norm of its rankings.

6.3.3 Sorting Algorithm

In the section the sorting of the possible solutions according to an associated ranking is explained. Here, a recursive quicksort algorithm is used [83]. The quicksort algorithm is a *divide-and-conquer algorithm*. It first divides the list to be sorted into two smaller sublists: the low elements and the high elements. Then, the algorithm can recursively sort the sublists [83, 84]. The quicksort algorithm used can be described in the following sequence.

- Step 1 *Pivot*: Select an element from the list (called a *pivot*).
- Step 2 *Partition*: Reorder the list so that all elements with values less than the *pivot* come before the *pivot*. Elements with values greater than the *pivot* come after it. Then, the *pivot* is in its final position.
- Step 3 *Sort 1*: Recursively sort the sublist of lesser elements.
- Step 4 *Sort 2*: Recursively sort the sublist of greater elements.

The algorithm is implemented using recursive functions written in `C` code [83]. In terms of operation numbers, the quicksort algorithm on average takes $n \log(n)$ comparisons to sort n elements. However, in the worst case, it makes n^2 comparisons. For example, if the number of elements to sort is $n = 8$, the average number of comparisons is only eight, while in the worst case, there are 64 comparisons. This is an important issue, as evidenced by more efficient algorithms in the literature [84]. Another possibility is to use parallel processing based on hardware implementations.

6.3.4 Implemented MPTC Algorithm

If an Euclidean-norm is considered, the minimization of (6.10) is performed through an exhaustive search for all feasible voltage vectors just as it is done in the standard PTC

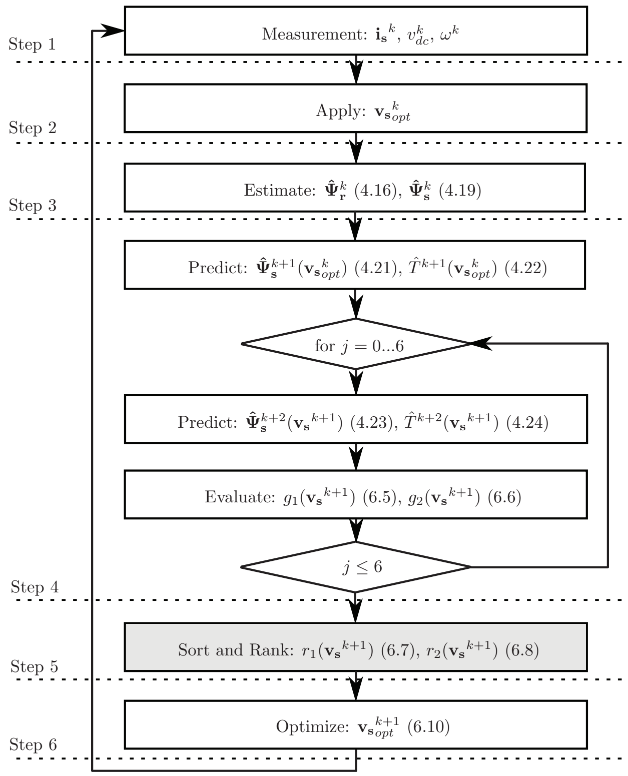


Figure 6.3: Multiobjective ranking-based predictive torque and flux control algorithm considering the calculation delay compensation.

approach. The proposed control strategy can be described in the following sequence (see Fig. 6.3).

- Step 1 *Measurement*: Sampling to get $\mathbf{i_s}^k$, v_{dc}^k and ω^k.
- Step 2 *Apply*: Set the optimal voltage vector $\mathbf{v_s}_{opt}^k$ found in the previous loop iteration.
- Step 3 *Estimate*: Flux estimations by using (4.16) and (4.19).
- Step 4 *Evaluate*: Extrapolate the control variables for $\mathbf{v_s}_{opt}^k$ using (4.21) and (4.22). Predict the control variables for every possible voltage vector $\mathbf{v_s}^{k+1}(j)$, with $j = 0, ..., 6$ using (4.23) and (4.24). Then, evaluate g_1 and g_2 using (6.5) and (6.6).
- Step 5 *Sort and Rank*: Sort the obtained values for each cost function from lower to higher. Assign a ranking value to each position using (6.7) and (6.8).
- Step 6 *Optimize*: Select optimal $\mathbf{v_s}_{opt}^{k+1}$ (minimization of (6.10)). Return to Step 1.

6.4 Fuzzy Decision-Making PTC (FPTC)

The multiobjective optimization problem to solve at each sampling time using fuzzy decision-making predictive torque control (FPTC) can be too stated as the fair minimization of the two different cost functions

$$g_1(\mathbf{v_s}^{k+1}) = \left(T^* - \hat{T}^{k+2}\right)^2, \tag{6.11}$$

$$g_2(\mathbf{v_s}^{k+1}) = \left(||\mathbf{\Psi_s}^*|| - ||\hat{\mathbf{\Psi}}_{\mathbf{s}}^{k+2}||\right)^2, \tag{6.12}$$

where $g_1(\mathbf{v_s}^{k+1})$ and $g_2(\mathbf{v_s}^{k+1})$ are the errors associated with the torque and stator flux, respectively. The proposed FPTC strategy evaluates these components separately for each possible voltage vector of the converter.

The operation of the fuzzy decision-making PTC strategy (FPTC) needs the specification of the membership functions and the type of confluence or decision. The used form of membership function is the linear, because results with quadratic and linear membership functions are the same. Then, the membership functions are defined as

$$\mu_1(\mathbf{v_s}^{k+1}) = \frac{g_1^{\max} - g_1(\mathbf{v_s}^{k+1})}{g_1^{\max} - g_1^{\min}}, \tag{6.13}$$

$$\mu_2(\mathbf{v_s}^{k+1}) = \frac{g_2^{\max} - g_2(\mathbf{v_s}^{k+1})}{g_2^{\max} - g_2^{\min}}, \tag{6.14}$$

where μ_1 and μ_2 are the membership function associated with torque and stator flux, respectively. Variables $g_1^{\max}$, $g_1^{\min}$, $g_2^{\max}$ and $g_2^{\min}$ are defined as,

$$g_1^{\max} = \max\{g_1(\mathbf{v_0}), \cdots, g_1(\mathbf{v_7})\}, \tag{6.15}$$

$$g_1^{\min} = \min\{g_1(\mathbf{v_0}), \cdots, g_1(\mathbf{v_7})\}, \tag{6.16}$$

$$g_2^{\max} = \max\{g_2(\mathbf{v_0}), \cdots, g_2(\mathbf{v_7})\}, \tag{6.17}$$

$$g_2^{\min} = \min\{g_2(\mathbf{v_0}), \cdots, g_2(\mathbf{v_7})\}. \tag{6.18}$$

Each membership function transform the errors from cost function to the range [0,1]. From the membership function definition, it can be seen that the mapping varies at each sampling time. In this way, the scale used for each cost function is variable.

6.4.1 Decision Function

The final selection of the best converter actuation is performed by the decision function. Two cases are implemented for FMPTC, the maximizing decision or `MIN` Operator defined by

$$\mu_D(\mathbf{v_s}^{k+1}) = \min\{\mu_1(\mathbf{v_s}^{k+1}), \mu_2(\mathbf{v_s}^{k+1})\}. \tag{6.19}$$

On the other hand, the Hamacher logical `AND` is defined by

$$\mu_D(\mathbf{v_s}^{k+1}) = \mu_1(\mathbf{v_s}^{k+1})\mu_2(\mathbf{v_s}^{k+1}). \tag{6.20}$$

where $\mu_1(\mathbf{v_s}^{k+1})$ and $\mu_2(\mathbf{v_s}^{k+1})$ are defined in (6.13) and (6.14), respectively.

Finally, the best converter actuation to be applied at the next sampling time is selected as the one with the maximum value of the decision function, thus

$$\mathbf{v_s}_{opt} = \arg \max_{\{\mathbf{v_0},\ldots,\mathbf{v_7}\}} \mu_D(\mathbf{v_s}^{k+1}). \tag{6.21}$$

6.4.2 Calculation Example

To illustrate the procedure, a calculation example is presented in Table 6.2. The example considers seven voltage vectors and two objective functions. First, the values g_1 and g_2 are calculated for each state with the help of the system model. Then, the maximum and minimum levels are determined for each objective obtaining $g_1^{\max} = 0.66$, $g_1^{\min} = 0.10$ and $g_2^{\max} = 0.0027$, $g_2^{\min} = 0.0001$. With these values, the membership functions μ_1, μ_2 can be calculated.

After that, the decision function μ_D is evaluated, where $\mu_D^{\texttt{MIN}}$ denotes the maximizing decision function and $\mu_D^{\texttt{AND}}$ denotes the product operator, respectively. Finally, the voltage vector is chosen; in this example, it corresponds to $\mathbf{v_3}$, for both decisions.

Table 6.2: State selection example

$\mathbf{v_s}$	$g_1(\mathbf{v_s})$	$g_2(\mathbf{v_s})$	$\mu_1(\mathbf{v_s})$	$\mu_2(\mathbf{v_s})$	$\mu_D^{\texttt{MIN}}$	$\mu_D^{\texttt{AND}}$
$\mathbf{v_0}$	0.10	0.0013	1	0.54	0.54	0.54
$\mathbf{v_1}$	0.60	0.0012	0.11	0.58	0.11	0.06
$\mathbf{v_2}$	0.33	0.0002	0.59	0.96	0.59	0.57
$\mathbf{v_3}$	0.31	0.0001	0.63	1	$\boxed{0.63}$	$\boxed{0.63}$
$\mathbf{v_4}$	0.36	0.0027	0.54	0	0	0
$\mathbf{v_5}$	0.27	0.0025	0.70	0.08	0.08	0.05
$\mathbf{v_6}$	0.66	0.0015	0	0.46	0	0.00

6.4.3 Implemented FPTC Algorithm

If a decision function based on maximization of μ_D is considered, the optimization is done by exhaustive evaluation for all feasible voltage vectors, as in the standard PTC approach. Based on Fig. 6.4, the control strategy can be described in the following sequence.

- Step 1 *Measurement*: Sampling to get $\mathbf{i_s}^k$, v_{dc}^k and ω^k.
- Step 2 *Apply*: Set the optimal voltage vector $\mathbf{v_s}_{opt}^k$ found in the previous loop iteration.
- Step 3 *Estimate*: Flux estimations by using (4.16) and (4.19).
- Step 4 *Evaluate*: Extrapolate the control variables for $\mathbf{v_s}_{opt}^k$ using (4.21) and (4.22). Predict the control variables for every possible voltage vector $\mathbf{v_s}^{k+1}(j)$, with $j = 0, ..., 6$ using (4.23) and (4.24). Then, evaluate g_1 and g_2 using (6.11) and (6.12).
- Step 5 *Fuzzification*: Using the g_1 and g_2 values, determine the maximum and minimum levels for each cost function at instant $k + 2$. Calculate μ_1 and μ_1 using (6.13) and (6.14) for every voltage vector.
- Step 6 *Optimize*: Obtain overall value μ_D for every voltage vector. Maximize the decision function using (6.19), and select optimal voltage vector $\mathbf{v_s}_{opt}^{k+1}$. Return to Step 1.

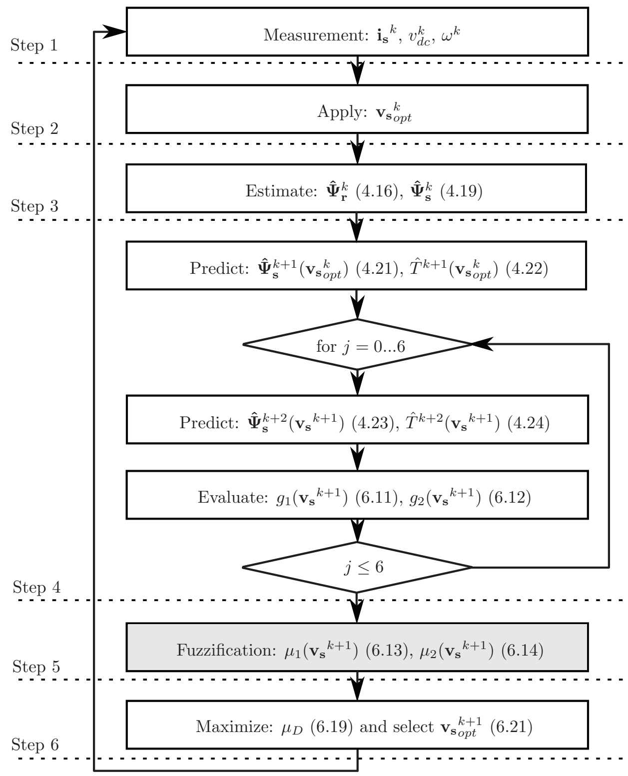


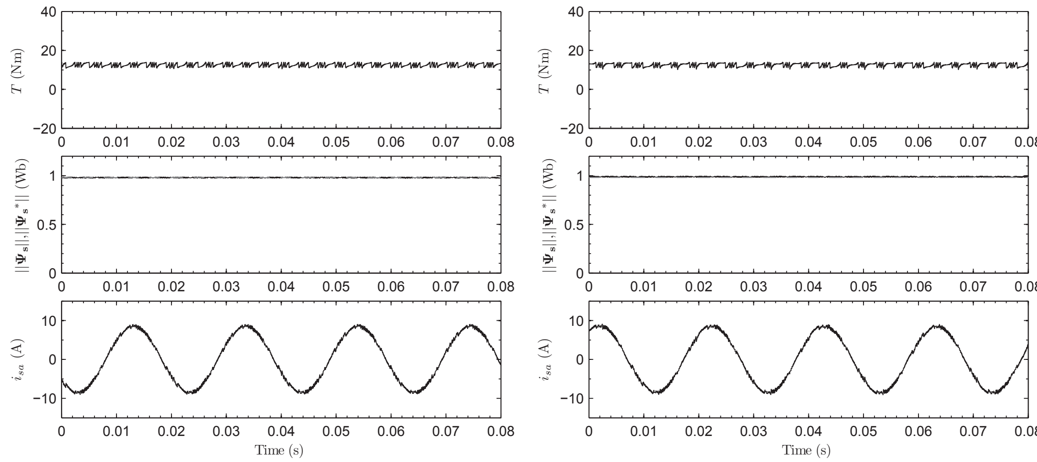
Figure 6.4: Fuzzy decision-making predictive torque and flux control algorithm considering the calculation delay compensation.

To show the dependence of the algorithm execution time on the implementation, the operation of the algorithm should be studied. The algorithm operates as follows. First, each cost function is evaluated for every switching state. Next, their maximum and minimum levels are obtained. Then, membership functions are calculated for every switching state, considering the previously determined levels. Finally, the decision is performed. As the evaluation of the cost functions and membership functions does not depend on each other, some stages are parallelizable. Thus, the calculation process could be divided in four sequential but internally parallel stages: cost function calculation, level determination, membership function evaluation, and decision.

6.5 Simulation Results

To validate the proposed multiobjective PTC schemes, a computer simulation using Matlab/Simulink has been performed using the parameters given in Table 4.2, which were selected according to the existing parameters of an experimental prototype. The control algorithms presented in Fig. 6.4 and Fig. 6.3 have been implemented in `C` because exactly the same code will be used for the experimental tests. The configuration of the simulations has been already presented in Chapter 4.

The implemented algorithms considers the switching scheme based on active vectors with zero-redundances illustrated in Fig. (4.5b) due to its simplicity and low stator current THD.



(a) (b)

Figure 6.5: Simulation results using the: (a) MPTC scheme; (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

The used approximation model is the Taylor modified because it corresponds to the best approximation to the continuous model according to the study presented in the Chapter 3.

Obviously, the cost functions g_1 and g_2 presented above for torque and stator flux, respectively, can be changed by

$$G(\mathbf{v_s}^{k+1}) = k_d g_1(\mathbf{v_s}^{k+1}) + k_q g_2(\mathbf{v_s}^{k+1}), \tag{6.22}$$

$$g_1(\mathbf{v_s}^{k+1}) = \left(i_{sd}^{*} - i_{sd}^{k+2}\right)^2 \tag{6.23}$$

$$g_2(\mathbf{v_s}^{k+1}) = \left(i_{sq}^{*} - i_{sq}^{k+2}\right)^2, \tag{6.24}$$

given arise to multiobjective predictive current control alternatives: Multiobjective Ranking-Based Predictive Current Control (MPCC) and Fuzzy Decision Predictive Current Control (FPCC). Finally, for the multiobjective ranking-based approach the Euclidean-norm is considered as an overall norm, while for schemes based on fuzzy decision-making the `MIN` operator is considered as decision function.

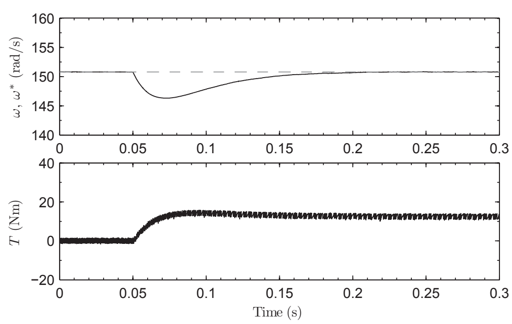
6.5.1 Strategies based on Multiobjective Ranking Approach

6.5.1.1 Steady-State Operation

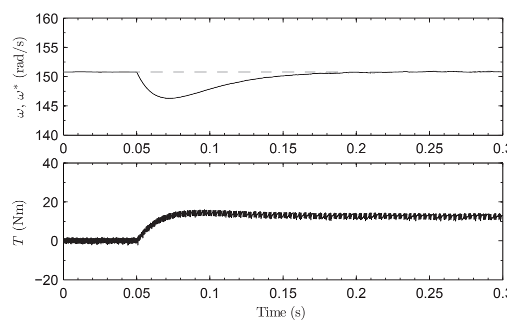
The first simulation presents the steady state behavior for the MPTC control strategy when the machine is operating at a nominal motoring speed at 1440 (rpm) with 50 (%) of the load torque, 12.5 (Nm). Fig. 6.5a shows the sinusoidal waveform of the stator current, then electric torque, and stator flux in steady state. The stator flux reference is the nominal one at 0.98 (Wb). The stator flux produces good current waveforms and, as a consequence, good torque performance in steady state. Fig. 6.5b presents the steady state behavior for the MPCC control strategy. The stator current in MPCC has low distortion, although similar

Table 6.3: Indices of different discrete-time models for MPTC and MPCC

Indices	Forward Euler	Matrix Fact.	Taylor	Taylor Modified
Multiobjective Ranking-Based PTC				
THD_{is} (%)	5.155	4.969	5.159	5.183
THD_{T} (%)	9.467	9.467	9.546	9.514
NRSMD_{Ψ_s} (%)	0.371	0.369	0.369	0.368
NRSMD_{T} (%)	6.685	6.659	6.741	6.755
$\tilde{f}_{sw}$ (kHz)	2.574	2.572	2.571	2.572
Multiobjective Ranking-Based PCC				
THD_{is} (%)	5.252	5.143	5.251	5.080
THD_{T} (%)	9.850	9.842	9.850	9.773
NRSMD_{Ψ_s} (%)	1.286	1.048	1.286	0.978
NRSMD_{T} (%)	7.511	7.447	7.511	7.242
$\tilde{f}_{sw}$ (kHz)	2.499	2.508	2.499	2.510



(a)



(b)

Figure 6.6: Simulation results using the: (a) MPTC scheme and (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Speed and torque behavior during a load impact of 50 (%) of the nominal load.

to MPTC. Finally, indices obtained with MPTC and MPCC, in steady state, are presented in Table 6.3.

6.5.1.2 Load Impact

The second result shows the performance of MPTC and MPCC under a load torque impact of 12.5 (Nm) while the machine is running at the nominal speed. In Fig. 6.6, the behavior of speed and torque is observed. Since the same PI-speed controller has been used, the rotor speed and torque reacts almost in the same way for both strategies.

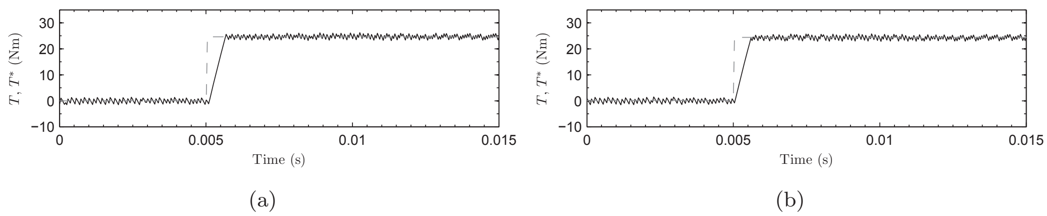


Figure 6.7: Simulation results using the: (a) MPTC scheme and (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Torque step response, (gray) torque reference, and (black) torque response.

6.5.1.3 Torque Response

The third simulation presents the dynamic behavior of the torque when the machine is operating at nominal flux condition. In Fig. 6.7, the torque step is equal to 25 (Nm), and it was performed by a change in the speed reference from 0.33(%) to 0.66(%) of ω_{nom}. A very quick transient response is obtained, due to the absence of an internal current control. Fig. 6.7a and Fig. 6.7b present the torque response for the MPTC and MPCC control strategy, respectively. Both schemes have the same torque response.

6.5.1.4 Speed Reversal Maneuver

Finally, the last simulation illustrates a speed-reversal operation from 0.9(%) to -0.9(%) of nominal speed. Fig. 6.8a shows the speed, torque, stator flux, and current waveform dynamics. The speed control is done with a PI controller, and for this reason, the speed has a smooth response. Furthermore, a decoupled control of the electric torque and the stator flux is achieved with the proposed MPTC method.

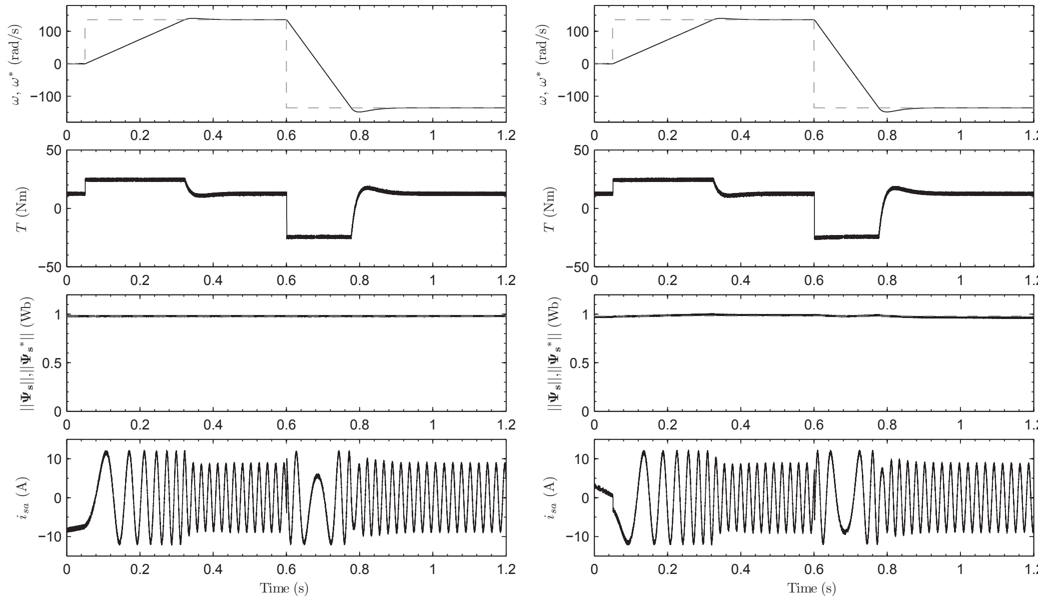
Fig. 6.8b shows the speed, torque, stator flux, and current waveform dynamics using the MPCC control strategy. Both schemes have similar dynamic responses and ripple, but the tracking of stator flux in MPTC is better than the obtained with MPCC. The above occurs because in MPCC the control variable is $\mathbf{i_s}_{dq}$, while in MPTC the control variables are stator flux and torque directly.

6.5.2 Strategies based on Fuzzy Decision-Making

6.5.2.1 Steady-State Operation

Similar to the above strategy, the first simulation result presents the steady state behavior for the FPTC control strategy when the machine is operating at a nominal motoring speed at 1440 (rpm) with 50 (%) of the load torque, 12.5 (Nm). Fig. 6.9a shows the sinusoidal waveform of the stator current, then electric torque, and stator flux in steady state. The stator flux reference is the nominal one at 0.98 (Wb). The stator flux produces good current waveforms and, as a consequence, good torque performance in steady state. Fig. 6.9b presents the steady state behavior for the FPCC control strategy.

The stator current in FPTC has low distortion, although similar to FPCC. Performance indices obtained with FPTC and FPCC, in steady state, are presented in Table 6.4. The above results are obtained using a `MIN` operator in the decision function. Similar values are



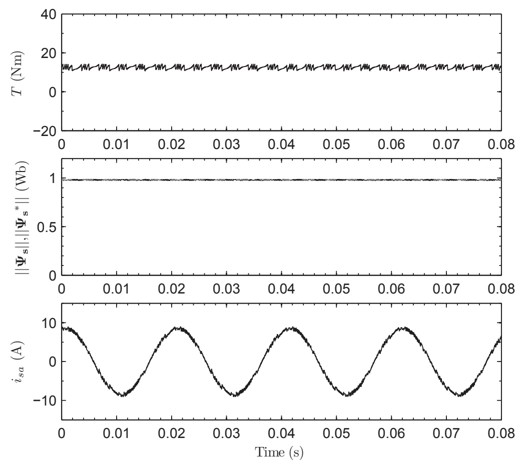
(a) (b)

Figure 6.8: Simulation results using the: (a) MPTC scheme and (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

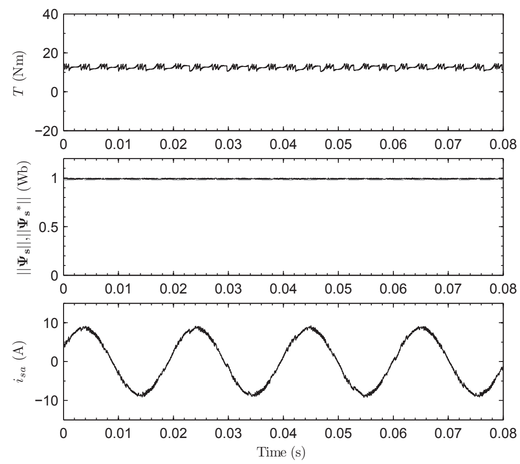
Table 6.4: Indices of different discrete-time models for FPTC and FPCC

Indices	Forward Euler	Matrix Fact.	Taylor	Taylor Modified
Fuzzy Decision-Making PTC				
THD_{is} (%)	5.075	5.249	5.131	5.214
THD_{T} (%)	9.599	9.649	9.891	9.661
NRSMD_{Ψ_s} (%)	0.375	0.374	0.372	0.371
NRSMD_{T} (%)	6.874	6.953	7.202	6.903
$\tilde{f}_{sw}$ (kHz)	2.571	2.573	2.560	2.561
Fuzzy Decision-Making PCC				
THD_{is} (%)	5.344	5.335	5.344	5.442
THD_{T} (%)	10.55	10.11	10.55	10.35
NRSMD_{Ψ_s} (%)	1.431	1.300	1.431	1.295
NRSMD_{T} (%)	9.441	9.149	9.441	9.136
$\tilde{f}_{sw}$ (kHz)	2.477	2.508	2.477	2.510

obtained with `AND` operator, they are presented in Table 6.5.



(a)



(b)

Figure 6.9: Simulation results using the: (a) FPTC scheme; (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

Table 6.5: Indices by using different decision function in FPTC

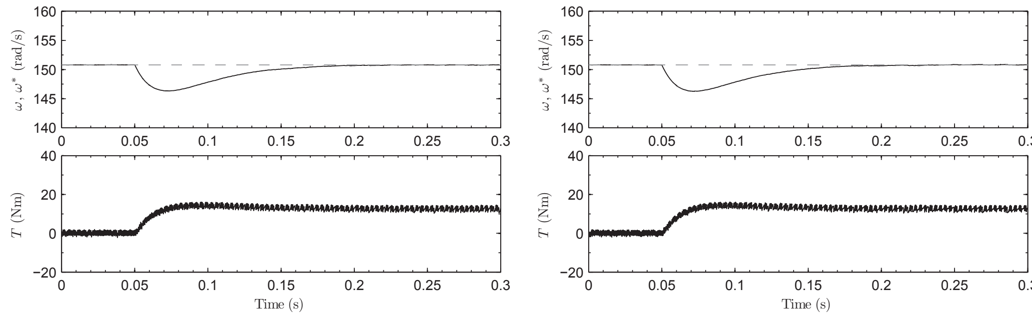
Indices	`MIN` Operator	`AND` Operator
THD_{is} (%)	5.214	5.210
THD_{T} (%)	9.661	9.484
NRSMD_{Ψ_s} (%)	0.371	0.374
NRSMD_{T} (%)	6.903	6.813
$\tilde{f}_{sw}$ (kHz)	2.561	2.573

6.5.2.2 Load Impact

The following result shows the performance of FPTC and FPCC under a load torque impact of 12.5 (Nm) while the machine is running at the nominal speed. In Fig. 6.11, the behavior of speed and torque is observed. Since the same PI-speed controller has been used, the rotor speed and torque dynamic are the same for both strategies.

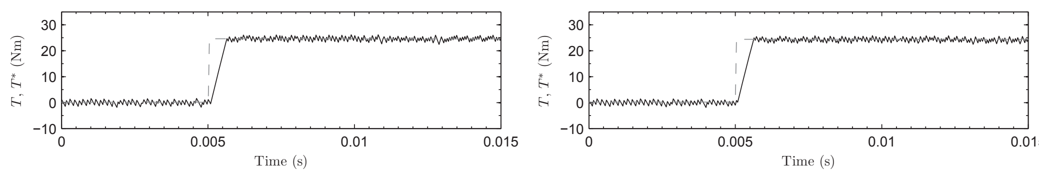
6.5.2.3 Torque Response

The next simulation presents the dynamic behavior of the torque when the machine is operating at nominal flux condition. In Fig. 6.11, the torque step is equal to 25 (Nm), and it was performed by a change in the speed reference from 0.33(%) to 0.66(%) of ω_{nom}. A quick transient response is obtained, due to the absence of an internal current control. Fig. 6.11a and Fig. 6.11b present the torque response for the FPTC and FPCC control strategy,



(a) (b)

Figure 6.10: Simulation results using the: (a) FPTC scheme and (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Speed and torque behavior during a load impact of 50 (%) of the nominal load.



(a) (b)

Figure 6.11: Simulation results using the: (a) FPTC scheme and (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Torque step response, (gray) torque reference, and (black) torque response.

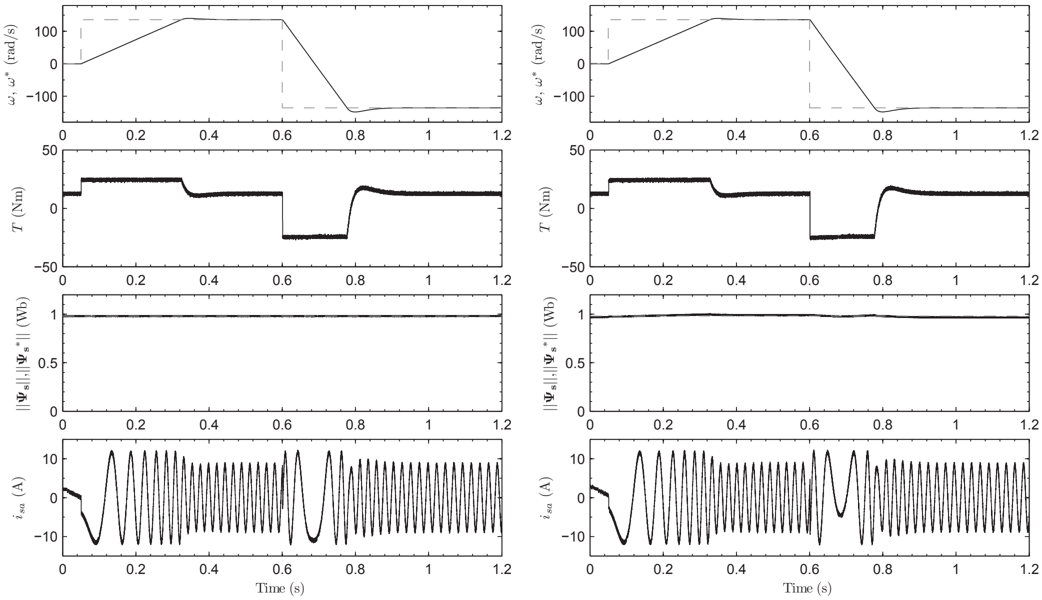
respectively. Finally, both schemes have the same torque response.

6.5.2.4 Speed Reversal Maneuver

The last simulation illustrates a speed-reversal operation from 0.9(%) to -0.9(%) of ω_{nom}. Fig. 6.12a shows the speed, torque, stator flux, and current waveform dynamics. The speed has a smooth response and a decoupled control of the electric torque and the stator flux is achieved with the proposed FPTC method. Fig. 6.12b shows the speed, torque, stator flux, and current waveform dynamics using the FPCC control strategy. Finally, both schemes have similar dynamic responses and ripple, but the tracking of stator flux in FPTC is better than the obtained with FPCC, as expected and reported in the above strategy.

6.6 Conclusions

Two new alternatives to solve the optimization process in the predictive torque scheme have been presented and applied successfully to the control of an induction motor drive. These strategies are based on methods from multiobjective optimization, and their main advantage is the avoidance of the weighting factors, and as consequence the elimination of the issue related with them to obtain adequate performance from the control scheme. The proposed



(a) (b)

Figure 6.12: Simulation results using the: (a) FPTC scheme and (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

control strategies have been validated through simulation, confirming from a theoretical point of view that the scheme has a good control performance.

From the comparison study with the standard PTC strategy a high performance is achieved. The transformation of the problem from the specification of weighting factors to the design of the multiobjective voltage selector does not seem as a big improvement. However, this application opens the possibility to developments in the field of multiobjective optimization to ease the design of the PTC state selector from a higher level and more natural description.

Finally, the presented simulation results indicate that the proposed algorithms allow good performance in PTC and PCC application. In contrast to the conventional approach, the proposed strategy does not require further parameter tuning for correct operation, and the optimized weighting-factor scheme is replaced by multiobjective stage. Several dynamic test have been provided, resulting in a good alternative to conventional approaches. Finally, in terms of steady state results, performance indices of ranking-based and fuzzy decision-making are similar in terms of stator current THD, instead of conventional weighted cost function scheme.

Chapter 7

EXPERIMENTAL VERIFICATION

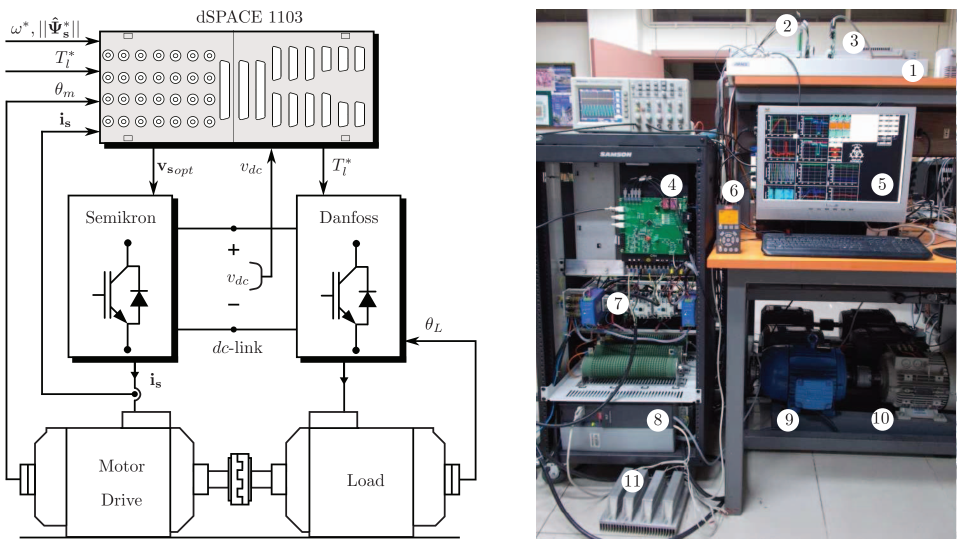
7.1 Introduction

APPLICATION of the FCS-MPC in Power Electronics has been tested and proven both theoretically and experimentally in the recent years. However, the implementation of FCS-MPC in the different power converters has given rise to some questions, such as the weighting factors calculation, frequency operation, long prediction horizon and steady-state error issues. The above chapters point out weighting factors calculation problem and its multiobjective optimization alternatives. They are simulated correctly, but experimental results are utilized to prove the effectiveness and usefulness of the proposed multiobjective methods.

7.2 Set-Up

The proposed strategy has been tested on an experimental rig, composed by two coupled induction machines. The motor drive is composed by a squirrel-cage induction machine controlled by a commercial three-phase two-level inverter, nominal power 20 (kW). A similar induction machine coupled to the same shaft using a semiflexible coupling is used as load. The load machine is fed by another commercial vector-controlled drive including rotor speed feedback for improved performance. The *dc*-link of both inverters are connected in parallel to achieve recirculation of power, avoiding the use of bulky braking resistors or regenerative rectifiers.

The set-up has been fully funded by FONDECYT, under grant N° 1100404, *High Performance Control of Electrical Machines*. The experimental setup diagram is presented in Fig. 7.1a, while Fig. 7.1b shows a photography of the mounted experimental rig. Basically, the experimental bench presented in Fig. 7.1b is composed by 1) dSPACE 1103 control platform, 2) I/Os, trips and PWM boards, 3) encoder board, 4) Semikron 2L-VSI Inverter, 5) Human Machine Interface (HMI), 6) load torque control panel, 7) measurement boards, 8) Danfoss inverter, 9) load machine, 10) controlled machine and 11) breaking resistors. Then, parts of this experimental bench are fully explained. The idea of using commercial inverters is to prove the proposed predictive control algorithms in typical industrial environment.



(a) (b)

Figure 7.1: Experimental set-up: (a) diagram; (b) rig.

7.2.1 Drive

The motor drive is composed by a squirrel-cage induction motor fed by a two level voltage source inverter (2L-VSI). Parameters of the motor, inverter, trips and measurement cards are explained here.

7.2.1.1 Induction Motor

The controlled machine is a squirrel-cage induction motor, model SIEMENS 1LA7113-4AA60, class IP55-112M B3, and its nominal power is 4 (kW). Parameters and nominal characteristics of this machine are presented in Table 7.1. The electrical parameters have been obtained through conventional test, no-load and blocked rotor tests. However, the used parameters are finally obtained with an industrial procedure offered by DANFOSS inverter. Motor values are set using the Automatic Motor Adaptation (AMA) in the control panel. This procedure excites the induction machine with different frequencies (high frequency) and it is the main advantage with regard to conventional tests, due to they are done with only one frequency (grid frequency). The method determines the motor settings, according to nominal parameters. The AMA function makes the setting without turning the motor shaft, which allows the motor to remain connected to the load during commissioning. A comparison between parameters are presented in Table 7.1. The set parameter obtained with AMA has been selected according to torque and speed torque response.

7.2.1.2 Inverter

The controlled inverter is a three-phase 2L-VSI, model SEMIKRON SKS 35F B6U+E1CIF+B6CI 21 V12, and its nominal power is 20 (kW). The main advantage of this model is the fully control of its gate signals. The firing pulses are commanded by a

Table 7.1: Parameters of controlled induction machine

Description	Parameter	Conventional Test	AMA Procedure
Electrical Parameters			
Stator Resistance	R_s	0.9667 (Ω)	1.6647 (Ω)
Rotor Resistance	R_r	2.0648 (Ω)	1.2134 (Ω)
Stator Inductance	L_s	162.53 (mH)	136.82 (mH)
Rotor Inductance	L_r	162.53 (mH)	136.82 (mH)
Magnetizing Inductance	L_m	154.81 (mH)	130.69 (mH)
Nominal Values			
Pair Poles	p	2	
Nominal Inertia	J	0.011 (kgm^2/s)	
Nominal Speed	ω_{nom}	1440 (rpm)	
Nominal Torque	T_{nom}	25 (Nm)	
Nominal Power	P_{nom}	4.0 (kW)	
Nominal Voltage	V_{nom}^{Δ}	400 (V)	

board incorporated into the inverter case. Some characteristics of this inverter are presented in Table 7.2. Notice that the inverter has IGBT SEMIKRON semiconductors based on three modules SK60GB128 for each leg, the breaking chopper is a module SK60GAL123, while the three phase diode bridge is a SK95D module.

The gate signals are commanded by an external board where active, negated signals and dead-times are implemented. Dead-time can be modified from 1, 2, 4 or 8 (μs). Other advantage of this external board is the incorporation of hardware-trip implementations, such as overcurrent V_{OC}, overvoltage V_{OV} and undervoltage V_{UV} protection, allowing load and converter safe operation. Undervoltage limit gives the value of when the forced-air cooled mechanism is activated. Hardware trips values are presented in Table 7.2. It can be noted that the input of this board are three gate signals and one external trip signal, each of them are transmitted by fiber-optic to avoid noise and electromagnetic interference.

7.2.1.3 Incremental Encoder

An incremental optical encoder with 4096 (ppr), model TURCK Ri-12H10T-2F4096-C-1M is used in both induction machines. The supply voltage of this encoder is from 5 to 30 (V), giving the possibility of direct connection from the digital encoder port (Inc 6, TTL-level compatible) of dSPACE platform. However, an adaptation board is preferred, increasing the supply voltage to avoid noise and electromagnetic interference in the position measurement used by the speed control loop. The board is a kind of voltage level transformation, providing a HTL to TTL transformation and insulation. Some features and terminal connections of this encoder and its adapter board are presented in Table 7.3.

7.2.1.4 Electrical Measurements

Despite that the command board mounted over the SEMIKRON inverter can measure two currents and one voltage, has been preferred the utilization of external measurement boards,

Table 7.2: SEMIKRON inverter parameters

Description	Parameter	Value
Inverter		
Max. Current	$I_o^{\max}$	35 (A)
Max. IGBT Voltage	$V_{CE}^{\max}$	1200 (V)
Max. Switching Frequency	$f_{sw}^{\max}$	15 (kHz)
Supply Voltage	V_s	380 (V)
dc-link Capacitor	C_{dc}	2040 (μF)
Max. *dc*-link Voltage	$V_{dc}^{\max}$	750 (V)
Firing Pulses board		
Selected Dead-time	-	2 (μs)
Voltage Supply	V_{cc}	24 (V)
Hardware Trips		
Overcurrent (Peak)	I_{OC}	16.752 (A) (0.698 (V)*)
Overvoltage (Peak)	I_{OV}	698.4 (V) (3.492 (V)*)
Undervoltage (Peak)	V_{UV}	449.0 (V) (2.245 (V)*)

* voltage in firing pulse board

Table 7.3: Incremental encoder parameters

Description	Parameter	Value
Encoder		
Voltage Supply Range	V_{cc}	[5,30] (V)
Incremental Resolution	-	4096 (ppr)
Starting Torque	-	0.05 (Nm)
Max. RPM	-	6000 (rpm)
Max. Low Signal	-	0.5 (V)
Max. High Signal	-	V_{cc}-1.0 (V)
Diameter	-	10 (mm)
Encoder Board		Color
Signal A	$A/\bar{A}$	Green/Yellow
Signal B	$B/\bar{B}$	Grey/Pink
Zero (Index) Z	$Z/\bar{Z}$	Blue/Red
V_{cc} and GND	V_{cc}/V_{GND}	Brown/White

because they are more noise-free. Furthermore, the measurement of three stator currents is chosen. The current board is based on LEM sensors model LAH-25NP, while voltage board is based on voltage dividers and instrumentation amplifiers, models INA121. The *dc* voltage supply of both board is 24 (V). Some features of electrical measurement board, gains and channel connections are presented in Table 7.4. Offsets values of measurements are adjusted according to operation conditions.

An important issue to be addressed in PTC implementation is the compensation of

Table 7.4: Measurements board parameters

Description	Parameter	Value
Current Board		
Voltage Supply	V_{cc}	24 (V)
Max. Current	$I_s^{\max}$	25 (A)
Current Gain ADC17, Phase a,	km_{Isa}	25.25 (A/V)
Current Gain ADC18, Phase b,	km_{Isb}	25.25 (A/V)
Current Gain ADC19, Phase c,	km_{Isc}	24.90 (A/V)
Voltage Board		
Voltage Supply	V_{cc}	24 (V)
Max. Voltage	$V_{dc}^{\max}$	750 (V)
dc-link Voltage Gain ADC20,	km_{Vdc}	801.5 (V/V)

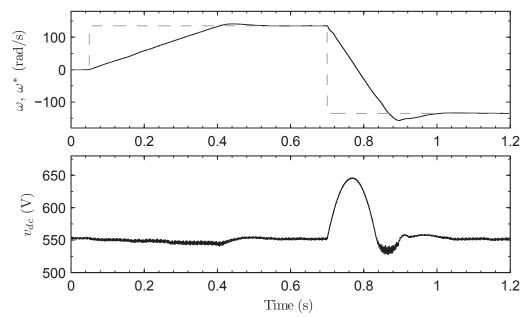


Figure 7.2: Speed and dc-link variation during a speed-reversal maneuver at 50(%) of the nominal load. Speed (top) and dc-link voltage (bottom).

dc-link voltage, such as oscillations due to the diode rectifier front-end or variations during breaking. Although these variations also affect the performance of linear controllers and can be compensated by feedforward, their effect is usually not critical in such implementations. On the other hand, predictions in PTC scheme are strongly dc-link voltage dependent, resulting in a problematic particulary in horizon-one alternatives. In the presented experimental results, it was found that the measurement of the dc-link voltage and its consideration in the prediction stage of the PTC algorithm significantly improves the quality of the practical results.

7.2.2 Load

7.2.2.1 Load Machine

The loadd machine is a squirrel-cage induction motor too, model WEG, class IP55-112M IEC60034, and its nominal power is 4 (kW). The parameters and nominal characteristics of this machine are presented in Table 7.5. The electrical parameters have been obtained through an industrial procedure offered by DANFOSS.

Table 7.5: Parameters of load induction machine

Description	Parameter	Value
Electrical Parameters		
Stator Resistance	R_s	1.2931 (Ω)
Rotor Resistance	R_r	0.8875 (Ω)
Stator Inductance	L_s	140.57 (mH)
Rotor Inductance	L_r	140.57 (mH)
Magnetizing Inductance	L_m	133.38 (mH)
Nominal Values		
Pair Poles	p	2
Nominal Inertia	J	0.01473 (kgm^2/s)
Nominal Speed	ω_{nom}	1440 (rpm)
Nominal Torque	T_{nom}	25 (Nm)
Nominal Power	P_{nom}	4.0 (kW)
Nominal Voltage	V_{nom}^{Δ}	380 (V)

7.2.2.2 Inverter

The load is fed by a three-phase 2L-VSI inverter, model DANFOSS FC302, and its nominal power is 11 (kW). This commercial inverter has two main advantages: its *dc*-link access and its breaking chopper circuit. The commissioning of this inverter is done in speed-control operation, while the load torque is commanded using vectorial control with or without encoder feedback. In our case, the control machine is used as machine load and then, the inverter must be configured in torque control option with an incremental encoder feedback of 4096 points connected to the shaft of the induction machine, in order to achieve better orientation and torque control. The most important features of the inverter are listed in Table 7.6.

In the test bench the motors are coupling, therefore the mechanical energy is transmitted through the mechanical axis. However, neither inverter has a regenerate capability, thus one alternative is the parallel connection of both *dc*-links, since the regenerative energy is circulating through both inverters. Excess energy during a braking operation are dissipated in resistors through a switch circuit incorporated in the inverter Danfoss. However, braking resistors used in the set-up are incorporated only for *dc*-links protection. Notice that the load torque reference is commanded by the user directly in the control panel of the inverter.

7.2.3 Control Platform

7.2.3.1 dSPACE 1103

The control platform used is a dSPACE, model ACE1103-PX4CLP-USB. The control strategies are programmed in `C`, while the HMI is programmed using ControlDesk software. The system has a primary processor for calculations, model PowerPC processor PPC750GX and another slave processor for data transmission and peripheral control, model DSP Texas Instruments TMS320F240. The system is connected to a desktop computer through a PCI slot. The control panel unit allows the connection of input and output signals, both analog

Table 7.6: DANFOSS inverter parameters

Description	Parameter*	Value
Inverter		
Max. Output Current	$I_o^{\max}$	24 (A)
Supply Voltage	V_s	380 (V)
Max. Input Current	$I_s^{\max}$	22 (A)
Max. *dc*-link Voltage	$V_{dc}^{\max}$	750 (V)
Parameters		
Output	1.00	Torque
Control Mode	1.01	Flux With Motor Feedback
Encoder	1.02	24 (V)
Encoder Feedback	1.06	Normal
Motor Data	1.2	Table 7.5
Advance Motor Data	1.3	Table 7.5
Min. Inertia	1.68	0.0119 (kgm^2/s)
Max. Inertia	1.68	0.0120 (kgm^2/s)
Breaking Mode	2.10	Resistor Brake
Brake Resistor	2.11	168 (Ω)
Brake Power Lim.	2.12	0.6 (kW)
Ref. Range	3.0	-MAX to +MAX
Min. Ref.	3.02	0 (Nm)
Max. Ref.	3.02	25 (Nm)
Ramp 1 Type	3.40	Linear
Ramp 1 Up Time	3.41	0 (s)
Ramp 1 Down Time	3.41	0 (s)

*parametes in control panel

and digital and digital encoders. The most important features used of the control platform are presented in Table 7.7.

7.2.3.2 Communication

The three gate signals and one external trip signal are transmitted by fiber-optic to avoid noise and electromagnetic interference. The signal are transmitted from dSPACE through I/O-PWM and I/O boards. The I/O-PWM is connected on the slave I/O PWM dSPACE port, while I/O board is connect on digital I/O dSPACE port. Both boards are fed by the control platform, but they can be externally supplied when more I/Os are required. The connection pins and software trips are summarized in Table 7.8.

7.2.3.3 HMI

The implemented HMI is developed using the software of dSPACE ControlDesk 3.7. The HMI is divided in three regions, Fig. 7.3. The first region is the graphical area, where variables are plotting with its references. In the next region are implemented the software trips, manual trip, controller states and the references. In this area there are three inputs,

Table 7.7: dSPACE parameters

Description	Value
Processors	
Primary Processor	PowerPC PPC750GX, float-point 64 bit
	CPU 1 GHz, 2x32kB L1, 1MB L2
	Bus 133MHz, 20 Interrupts
	Local SDRAM 32MB, Global SDRAM 96MB
Slave Processor	Texas Instruments TMS320F240
	fixed-point 32 bit, CPU 20 MHz
Input/Outputs	
Parallel ADCs	Channel 17-20, 16 bit, ±10 (V)
	Conv. Time 800 (ns)
Muxed ADCs	Channel 01-16, 16 bit, ±10 (V)
	Conv. Time 1000 (ns)
DACs	Channel 1-8, 16 bit, ±10 (V)
Digital I/Os	32 bit, TTL
Peripheral	
Timer	Timer A, B, 32 bit
PWM	1 3ϕ-PWM, 4 1ϕ-PWM outputs
Digital Encoder	6 Incremental Encoder, CH1-6
	TTL 5 (V)/1.5 (A), 24-bit
Analog Encoder	1 Incremental Encoder, CH7
	TTL 5 (V)/1.5 (A), 6-bit

Table 7.8: I/O board parameters

Description	Parameter	Value
I/O board		
Software Trip	Trip	Port IO11, 0x0000800
Gate Signal Phase a	S_a	Port IO10
Gate Signal Phase b	S_b	Port IO09
Gate Signal Phase c	S_c	Port IO08
I/O-PWM board		
Gate Signal Phase a	S_a	Port SPWM1
Gate Signal Phase b	S_b	Port SPWM3
Gate Signal Phase c	S_c	Port SPWM5
Software Trips		
Overcurrent (Peak)	I_{OC}	17 (A)
Overvoltage (Peak)	V_{OV}	700 (V)

stator flux magnetization (0.98 (Wb) or 7.115 (A) depending of the strategy), speed reference and weighting factor. The controller states are two: the inner predictive controller and the

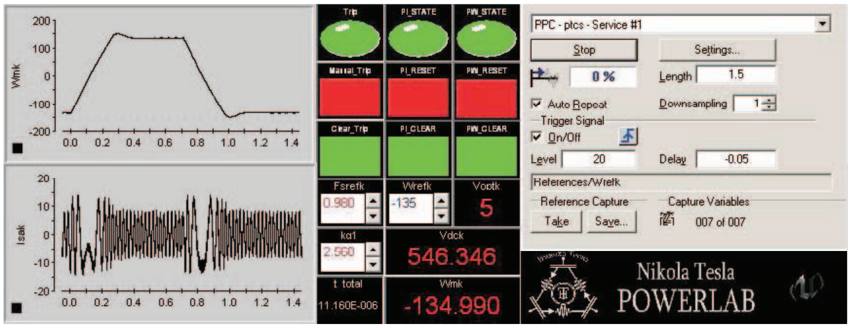


Figure 7.3: Implemented HMI. Plotting area (right), trips, controllers and references area (center), synchronization and capture area (left).

Table 7.9: Test configurations

Trigger	Level	Delay	Time
Steady-State Operation			
Speed reference, ω^*	0	-0.1 (s)	0.5 (s)
Load Impact			
Estimated torque, $\hat{T}_k$	5 (Nm)	-0.1 (s)	0.6 (s)
Torque Response			
Speed reference, ω^*	100 (rads/s)	-0.1 (s)	0.5 (s)
Speed Reversal Maneuver			
Speed reference, ω^*	20 (rads/s)	-0.1 (s)	1.5 (s)

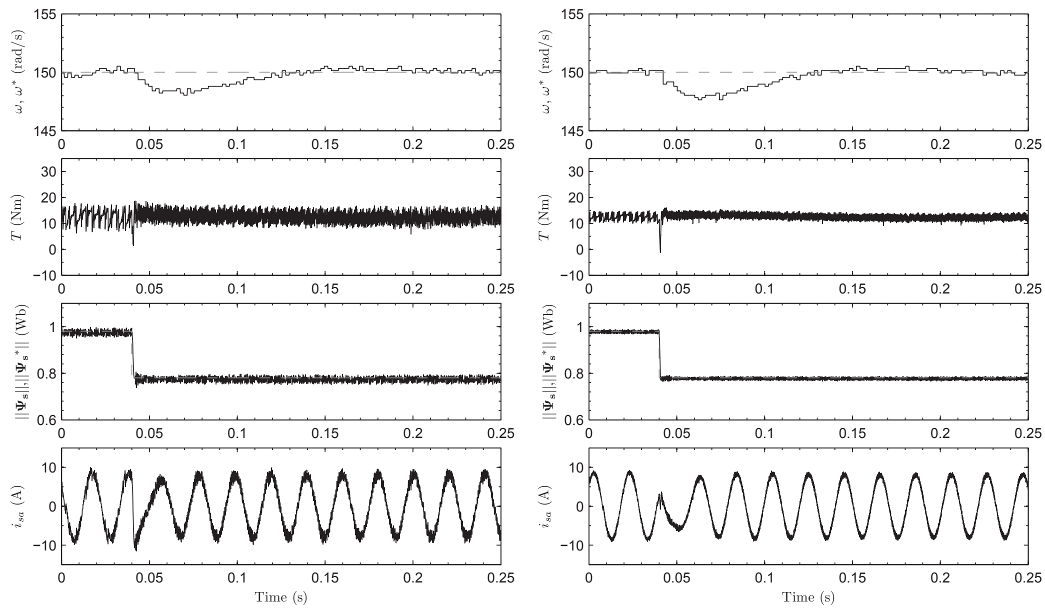
external speed controller. Finally, the last final area is used for synchronization with a determined variable and data capture settings. The trigger and capture settings of each test are illustrated in Table 7.9.

7.3 Experimental Results

This section describes the most important aspects of the experimental implementation for each of the control strategies studied in the previous chapters. Experimental results are presented to verify and compare the performance of the strategies in different operating points.

7.3.1 Scheme Configurations

The used sampling time has been selected according to stator current THD results, e.g., by using 100 (μs) and 40 (μs), the obtained current THD values are 12.15 (%) and 4.264 (%), respectively. The consideration for the selected sampling time is 5 (%) of current THD, considering that less distortion of stator currents extends the life of the machine. As expected, a reduction on the sampling time of the control system causes an increment on the average switching frequency, e.g., in the above case the average switching frequency



(a) (b)

Figure 7.4: Experimental flux weakening test of FPTC at: (a) 100 (μs) and (b) 40 (μs), in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

is increased from 0.94 (kHz) to 2.242 (kHz). Fig. 7.4 shows a general difference between the above sampling times by using the multiobjective fuzzy decision-making predictive torque control (FPTC) under a flux weakening test from 100 (%) to 80 (%) of stator flux operating with to nominal speed and 50 (%) of load torque.

The following point describes the general configuration of the experimental implementation of each of the studied strategies. Programming structure used to implement each strategy is based on the use of timer interrupts. In the case of Predictive Torque Control and Multiobjective approaches, they are using the Timer A interrupt, which is set at 40 (μs). In general, the settings used for experimental results are the same than presented in Table 4.2 for simulation results.

The software trip logic of the each algorithm is a very important aspect, since any measure helps to protect both equipment and users. The more important implemented logic is activating a trip signal by the overcurrent in any of the phases of the stator of the machine to be controlled. The maximum limit for this quantity is 17 (A), Table 7.8. Thus, if any stator current phase reached this value, a digital signal disables the firing pulses card connected to the controlled inverter.

The external speed controller designed in discrete-time using a ZOH discretization and implemented with anti-windup in order to limit the value of actuation (quadrature current) and then of the torque reference. Furthermore, the speed control loop is implemented at a subsampled rate of 1000 (Hz) in order to reduce the quantization error in the speed signal derived from the incremental encoder. It can be shown that the quantization error of the velocity is inversely proportional to the sampling time. Therefore, for very small sampling time, quantization errors are rather large, which creates very vigorous action on the output of the speed controller, resulting in a large amount of control distortion. While the angle of the encoder is obtained at the same rate that the currents of the machine, thus calculating the speed and performance of the speed PI control are performed at a frequency of 1000 (Hz) in all strategies.

7.3.2 Conventional Predictive Schemes

The conventional predictive schemes implemented here are two: predictive torque control (PTC) and the predictive current control (PCC). Both schemes have been tested under the same tests presented for simulation results in Chapter 4, such as steady-state operation, load impact, torque response and speed reversal operation. The used discrete-time model for following figures is the named Taylor modified due to lower theoretical error; however, major details about it are presented in a discussion section.

7.3.2.1 Steady-State Operation

The experimental optimal weighting factor used in PTC is the same of simulation results, $k_\Psi = 4096$ (normalized $\lambda = 2.56$), while for PCC the experimental optimal weighting factor used is $k_d = 0.61$. The selection of this value is explained in the a discussion section.

The first test shows the performance of the PTC and PCC strategies in steady state. The selected operation point considers a nominal rotor speed with a load torque $T_l = 12.5$ (Nm). Fig. 7.5 presents the stator current, stator flux, and torque behavior in steady state. It can be observed that, the current presents a typical switched waveform with very low distortion. The stator current in PCC presents low distortion and ripple compared with PTC. The distortion of the stator currents, the stator flux, and the torque ripple have been calculated in Table 7.10.

From Table 7.10, the lower stator current THD is obtained with matrix factorization discretization method; however, the lower torque THD is obtained with Taylor modified discretization method. For this reason, both models are good discretization alternatives. If more performance indices are considered, the analysis is a bit more complicated. This issue is pointed out in a weighting factor discussion section.

In terms of stator current THD and weighting factor complexity the PCC strategy presents an advantage with respect to PTC. However, PCC has a high torque ripple compared to PTC. Finally, the average switching frequency for both strategies are equivalent, it is around 2.7 (kHz) for each insulated gate bipolar transistor.

7.3.2.2 Load Impact

The following experimental result shows the performance of PTC and PCC under a load torque impact of 12.5 (Nm) while the machine is running at the nominal speed. In Fig. 7.6, the behavior of the speed and torque is observed. Since the same PI-speed controller has been used, the rotor speed and torque reacts almost in the same way for both strategies.

7.3.2.3 Torque Response

The next experimental result shows the dynamic behavior of the torque when the machine is operating at nominal flux condition. In Fig. 7.7, the torque step is equal to 25 (Nm), and

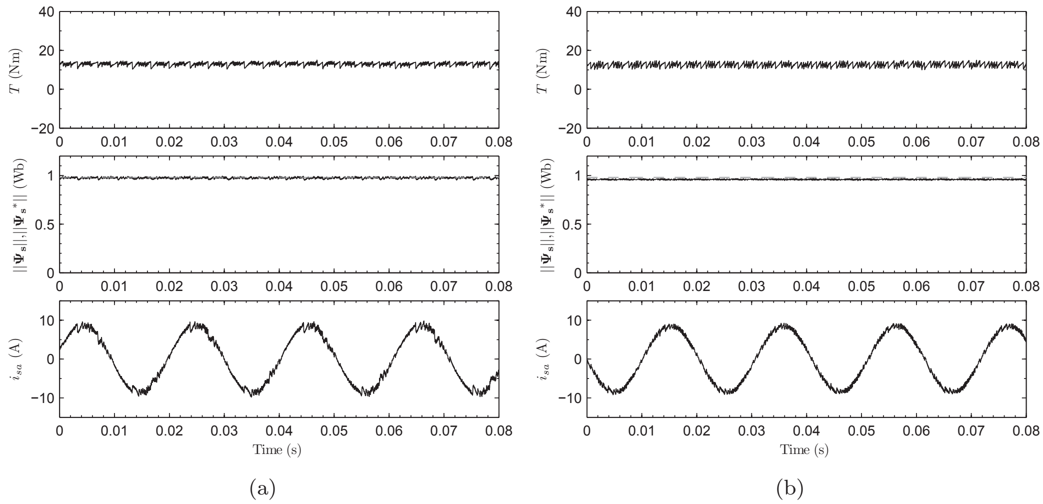
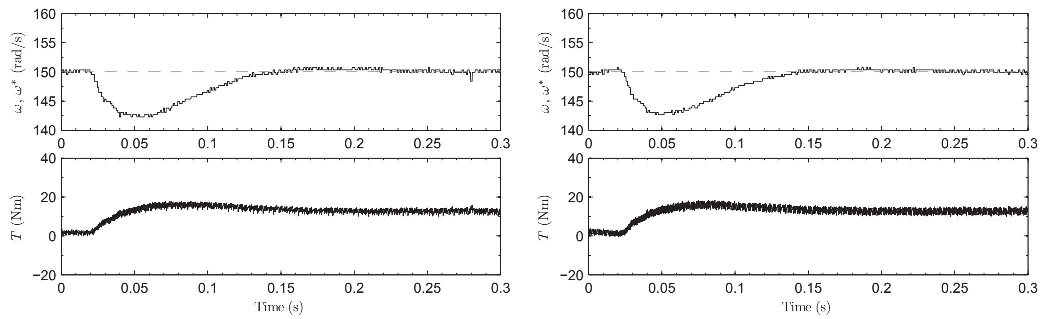


Figure 7.5: Experimental results using the: (a) PTC scheme; (b) PCC scheme, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

Table 7.10: Experimental indices of PTC and PCC

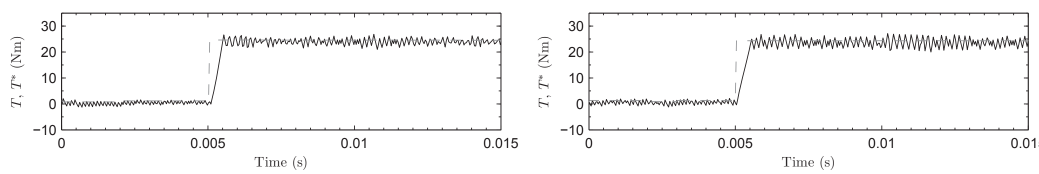
Indices	Forward Euler	Matrix Fact.	Taylor	Taylor Modified
Predictive Torque Control				
THD_{is} (%)	6.657	6.550	6.782	6.731
THD_T (%)	7.305	7.402	7.390	7.119
NRSMD_{Ψ_s} (%)	0.842	0.836	0.849	0.864
NRSMD_T (%)	6.554	6.634	6.599	6.534
$\tilde{f}_{sw}$ (kHz)	2.803	2.796	2.777	2.813
Predictive Current Control				
THD_{is} (%)	4.104	4.011	4.036	3.996
THD_T (%)	8.631	8.613	8.681	8.493
NRSMD_{Ψ_s} (%)	0.504	0.510	0.510	0.510
NRSMD_T (%)	8.416	8.353	8.455	8.381
$\tilde{f}_{sw}$ (kHz)	2.598	2.627	2.635	2.632

it was performed by a change in the speed reference from 0.33(%) to 0.66(%) of nominal speed (from 50 to 100 (rad/s)). A quick transient response is obtained due to the absence of an internal current control. Fig. 7.7a and Fig. 7.7b present the torque response for the PTC and PCC control strategy, respectively. Both schemes have the same torque response; however, the torque ripple is higher in PCC as expected.



(a) (b)

Figure 7.6: Experimental results using the: (a) PTC scheme and (b) PCC scheme, in an induction machine fed by a 2L-VSI. Speed and torque behavior during a load impact of 50 (%) of the nominal load.



(a) (b)

Figure 7.7: Experimental results using the: (a) PTC scheme and (b) PCC scheme, in an induction machine fed by a 2L-VSI. Torque step response, (gray) torque reference, and (black) torque response.

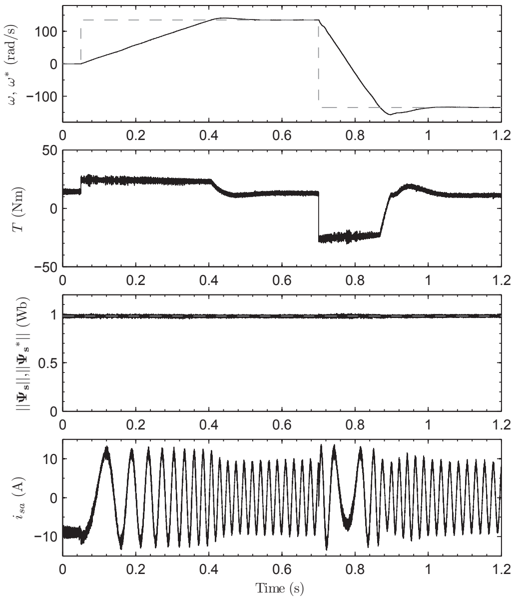
7.3.2.4 Speed Reversal Maneuver

Finally, the last experimental results of conventional schemes illustrate a speed-reversal operation from 0.9(%) to -0.9(%) of ω_{nom}. Fig. 7.8a shows the speed, torque, stator flux, and current waveform dynamics. The speed has a smooth response. It is not the case of the torque and the stator flux due to the use of a nonlinear control law. Finally, a decoupled control of the electric torque and the stator flux is achieved with the proposed PTC method.

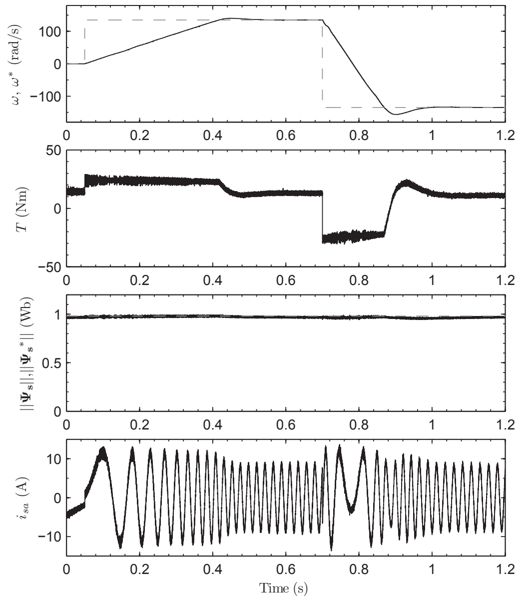
Fig. 7.8b shows the speed, torque, stator flux, and current waveform dynamics using the PCC control strategy. Both schemes have similar dynamic responses; however, the torque ripple of PTC is lower compared to PCC, but the tracking of stator flux in PTC is better than the obtained with PCC. In facts, in PCC the control variable is $\mathbf{i_s}_{dq}$, while in PTC the control variables are stator flux and torque directly.

7.3.3 Strategies based on Multiobjective Ranking Approach

An Euclidean-norm is considered as an overall norm in the proposed multiobjective ranking-based approaches. The proposed MPTC and MPCC schemes have been tested under the same tests presented for simulation results in Chapter 6, such as steady-state operation, load impact, torque response and speed reversal operation. The used discrete-time model



(a)



(b)

Figure 7.8: Experimental results using the: (a) PTC scheme and (b) PCC scheme, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

is Taylor modified approximation as was used above.

7.3.3.1 Steady-State Operation

The first experimental result presents the steady state behavior for the MPTC control strategy when the machine is operating at a nominal motoring speed at 1440 (rpm) with 50 (%) of the load torque, 12.5 (Nm). Fig. 7.9a shows the sinusoidal waveform of the stator current, then electric torque, and stator flux in steady state. The stator flux reference is the nominal one at 0.98 (Wb). The stator flux produces good current waveforms and, as a consequence, good torque performance in steady state.

Fig. 7.9b presents the steady state behavior for the MPCC control strategy. The stator current in MPCC has low distortion, although similar to MPTC. Performance indices obtained with MPTC and MPCC, in steady state, are presented in Table 7.11. Compared with conventional schemes, multiobjective strategies have lower stator current THD operating under a more lower switching frequency. In fact, in MPTC there is an important reduction on stator current THD and average switching frequency compared with conventional PTC. Similar results are obtained with MPCC.

7.3.3.2 Load Impact

The next experimental result shows the performance of MPTC and MPCC under a load torque impact of 12.5 (Nm) while the machine is running at the nominal speed. In Fig.

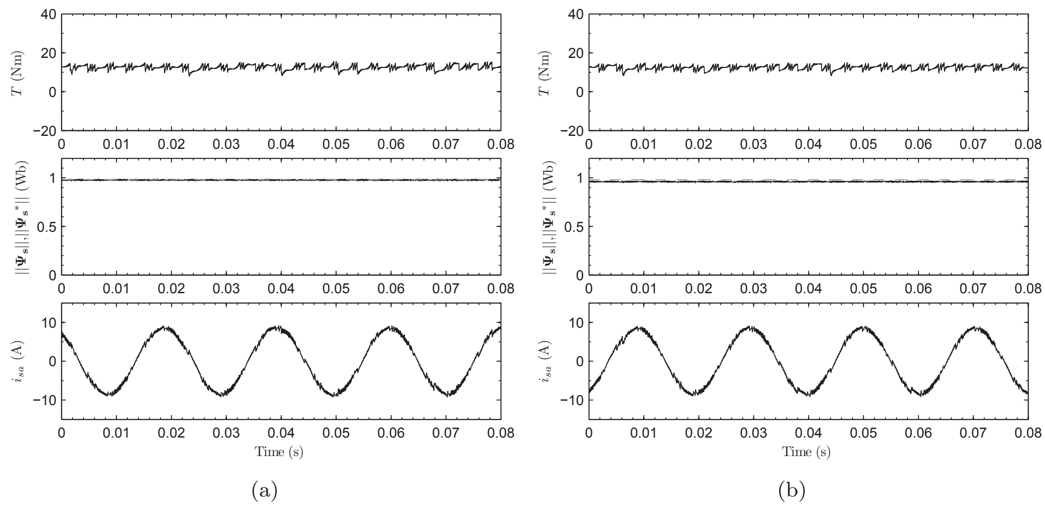
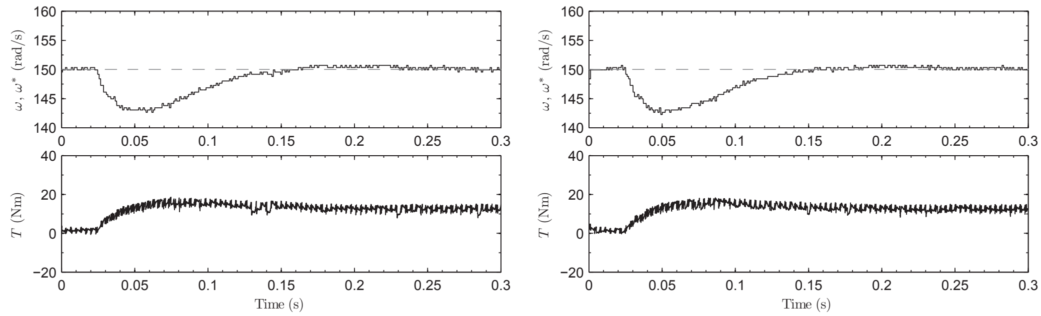


Figure 7.9: Experimental results using the: (a) MPTC scheme; (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

Table 7.11: Experimental indices of MPTC and MPCC

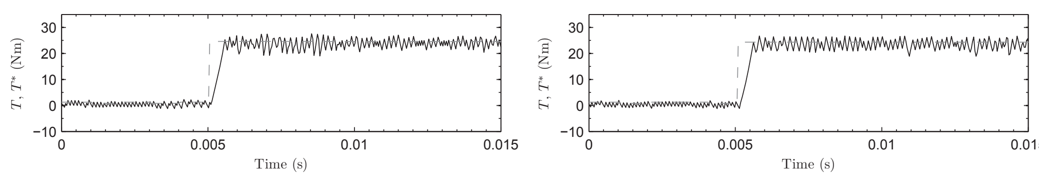
Indices	Forward Euler	Matrix Fact.	Taylor	Taylor Modified
Multiobjective Ranking-Based PTC				
THD_{is} (%)	4.951	4.864	4.919	5.055
THD_{T} (%)	12.53	12.26	11.89	11.65
NRSMD_{Ψ_s} (%)	0.457	0.457	0.459	0.454
NRSMD_{T} (%)	10.250	10.040	9.831	9.520
$\tilde{f}_{sw}$ (kHz)	2.359	2.348	2.363	2.300
Multiobjective Ranking-Based PCC				
THD_{is} (%)	4.526	4.330	4.187	4.297
THD_{T} (%)	10.39	10.17	9.867	10.23
NRSMD_{Ψ_s} (%)	0.456	0.457	0.457	0.454
NRSMD_{T} (%)	8.726	8.700	8.492	8.718
$\tilde{f}_{sw}$ (kHz)	2.323	2.358	2.337	2.337

7.10, the behavior of the speed and torque is observed. Since the same PI-speed controller has been used, the rotor speed and torque reacts almost in the same way for both and conventional strategies. Note that, the torque ripple is higher compared with conventional schemes.



(a) (b)

Figure 7.10: Experimental results using the: (a) MPTC scheme and (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Speed and torque behavior during a load impact of 50 (%) of the nominal load.



(a) (b)

Figure 7.11: Experimental results using the: (a) MPTC scheme and (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Torque step response, (gray) torque reference, and (black) torque response.

7.3.3.3 Torque Response

The follow experimental result presents the dynamic behavior of the torque when the machine is operating at nominal flux condition. In Fig. 7.11, the torque step is equal to 25 (Nm), and it was performed by a change in the speed reference from 0.33(%) to 0.66(%) of nominal speed. A quick transient response is obtained, due to the absence of an internal current control. Fig. 7.11a and Fig. 7.11b present the torque response for the MPTC and MPCC control strategy, respectively. Both schemes have the same torque response. In terms of torque ripple, both strategies presents a higher value compared with conventional strategies. The high ripple appears in a no-load operation due to the control actuation is low.

7.3.3.4 Speed Reversal Maneuver

Finally, the last experimental test illustrates a speed-reversal operation from 0.9(%) to -0.9(%) of nominal speed. Fig. 7.12a shows the speed, torque, stator flux, and current waveform dynamics. The speed has a smooth response and a decoupled control of the electric torque and the stator flux is achieved with the proposed MPTC method without any weighting factor selection or *a priori* consideration for each objective. Notice that the

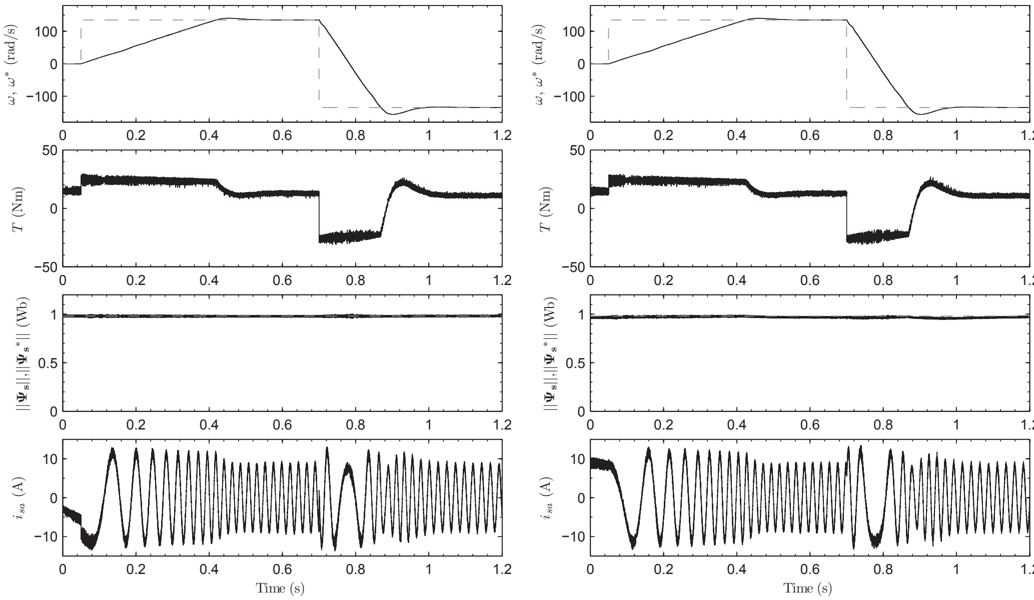


Figure 7.12: Experimental results using the: (a) MPTC scheme and (b) MPCC scheme, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

proposed scheme considers that the objectives should be fairly optimized in each sampling time.

7.3.4 Strategies based on Fuzzy Decision-Making

Schemes based on fuzzy decision-making are validated here. The MIN operator is considered as decision function. The proposed FPTC and FPCC have been tested under the same tests presented for simulation results in Chapter 6 and experimental configurations presented above.

7.3.4.1 Steady-State Operation

The first test shows performance of the FPTC and FPCC strategies in steady state. The selected operation point considers a nominal rotor speed with a load torque $T_l = 12.5$ (Nm). Fig. 7.13 presents the stator current, stator flux, and torque behavior in steady state. It can be observed that, stator current presents a typical switched waveform with very low distortion. Stator current in FPCC presents low distortion and ripple compared with FPTC. The distortion of the stator currents, the stator flux ripple, and torque ripple have been calculated in Table 7.12. In terms of stator current THD and weighting factor complexity the proposed FPTC and FPCC strategies present an advantage with respect to conventional schemes. Finally, average switching frequency is the same for both strategies, it is around 2.2 (kHz) for power switch.

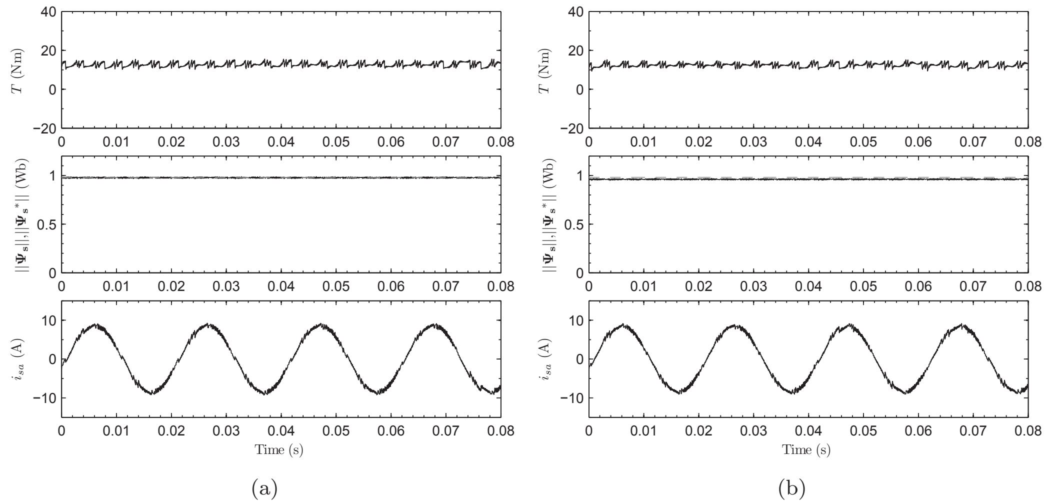


Figure 7.13: Experimental results using the: (a) FPTC scheme; (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

Table 7.12: Experimental indices of FPTC and FPCC

Indices	Forward Euler	Matrix Fact.	Taylor	Taylor Modified
Fuzzy Decision-Making PTC				
THD_{is} (%)	4.498	4.249	4.422	4.264
THD_{T} (%)	10.62	10.28	10.61	10.27
NRSMD_{Ψ_s} (%)	0.447	0.449	0.448	0.448
NRSMD_{T} (%)	8.688	8.493	8.647	8.491
$\tilde{f}_{sw}$ (kHz)	2.269	2.275	2.235	2.242
Fuzzy Decision-Making PCC				
THD_{is} (%)	4.492	4.176	4.498	4.443
THD_{T} (%)	10.73	10.01	10.79	10.52
NRSMD_{Ψ_s} (%)	0.460	0.451	0.452	0.450
NRSMD_{T} (%)	8.716	8.335	8.791	8.637
$\tilde{f}_{sw}$ (kHz)	2.310	2.268	2.288	2.274

7.3.4.2 Load Impact

The next experimental result shows the performance of FPTC and FPCC under a load torque impact of 12.5 (Nm) while the machine is running at nominal speed. In Fig. 7.14, behavior of speed and torque is observed. Since the same PI-speed controller has been used, rotor speed and torque reacts almost in the same way for both strategies.

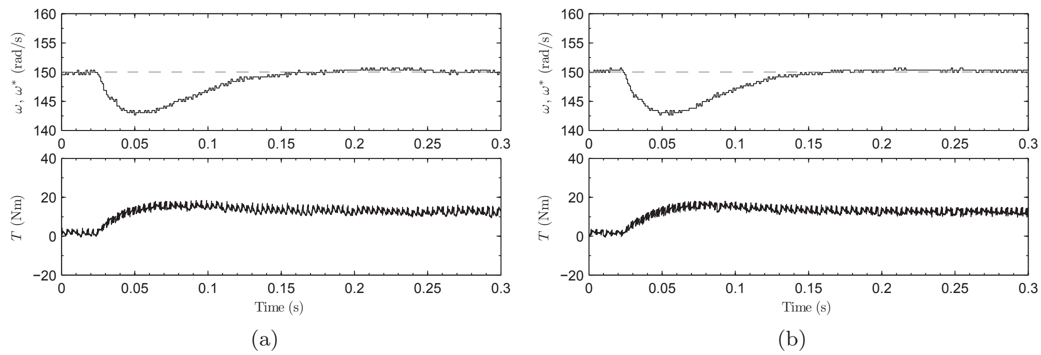


Figure 7.14: Experimental results using the: (a) FPTC scheme and (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Speed and torque behavior during a load impact of 50 (%) of the nominal load.

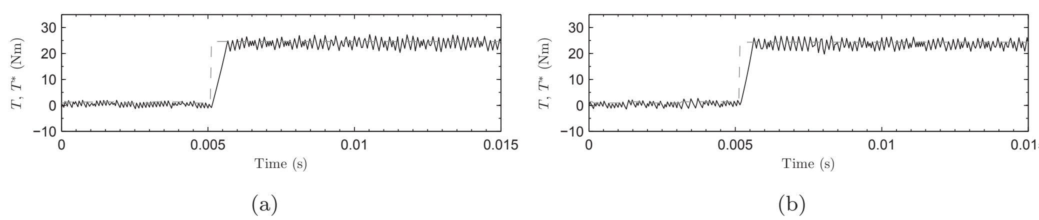


Figure 7.15: Experimental results using the: (a) FPTC scheme and (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Torque step response, (gray) torque reference, and (black) torque response.

7.3.4.3 Torque Response

The following experimental results shows the dynamic behavior of the torque when the machine is operating at nominal flux condition (0.98 (Wb)). In Fig. 7.15, the torque step is equal to 25 (Nm), and it was performed by a change in the speed reference from 0.33(%) to 0.66(%) of nominal speed (from 50 to 100 (rad/s)). A quick transient response is obtained. Fig. 7.15a and Fig. 7.15b present torque response for FPTC and FPCC control strategy, respectively. Notice that both schemes have the same torque dynamical response and ripple.

7.3.4.4 Speed Reversal Maneuver

Finally, the last experimental results illustrates a speed-reversal operation from 0.9(%) to -0.9(%) of nominal speed. Fig. 7.16 shows the speed, torque, stator flux, and current waveform dynamics. Speed has a smooth response. It is not the case of the torque and the stator flux due to the use of a nonlinear control law. Finally, a decoupled control of the electric torque and the stator flux is achieved with proposed methods. Fig. 7.16b shows the speed, torque, stator flux, and current waveform dynamics using the FPCC control strategy. Both schemes have similar dynamic responses and ripple.

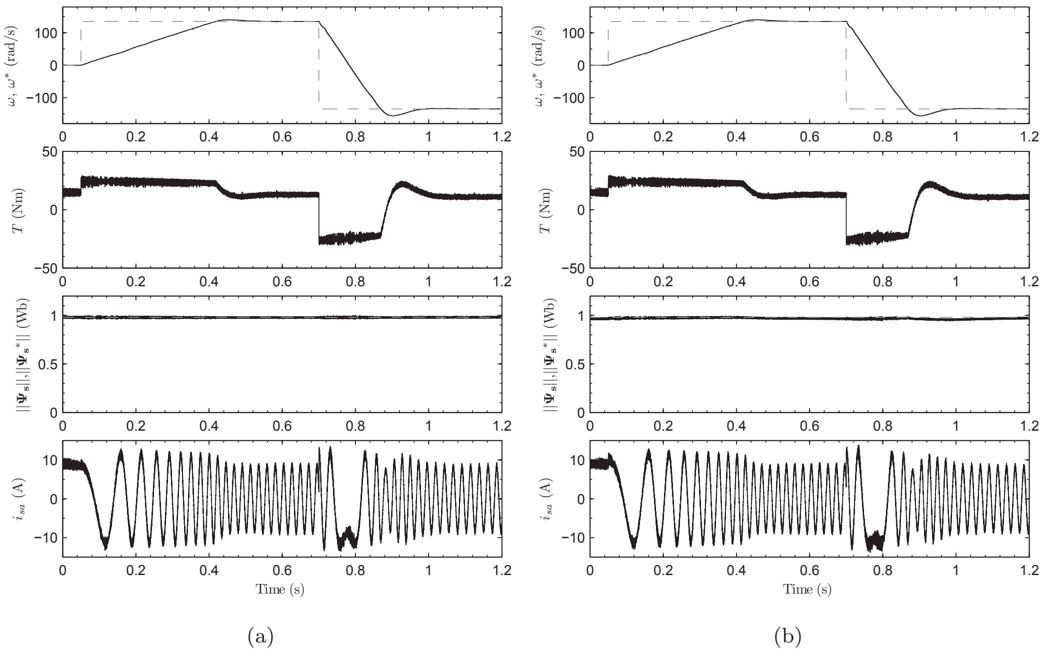


Figure 7.16: Experimental results using the: (a) FPTC scheme and (b) FPCC scheme, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

7.4 Discussion

In this section some discussions of experimental results are presented. Each strategy and its relevant implementation issues are analyzed and commented.

7.4.1 Model Discussion

The determination of the best discrete-time model is done assuming sufficient computational power, for this reason only merit functions in state-state of each presented strategy are considered. The strategies used are optimal PTC and PCC, MPTC, MPCC, FPTC and FPCC, while the merit functions are THD_{is}, THD_T, NRSMD_{Ψ_s}, NRSMD_T and $\tilde{f}_{sw}$, e.g., for THD_{is}, the discrete-time model based on matrix factorization gives the best value for all considered strategies, thus, the rank associated is the lowest (rank 1), while the worst case is for forward euler (rank 4), (see Table 7.13). Now, if the same procedure is performed with another merit function, the best overall discrete-time model is the Taylor modified. However, as expected matrix factorization is a good alternative.

As is well-known, the machine parameters can be determined by conventional tests (locked rotor and no-load), in which the machine is directly connected to the sinusoidal grid. However, in the above experimental results the parameters have been obtained with an commercial procedure. Finally, in PTC, switching frequency is both higher and variable, and parameters change at such a high frequency. Hence, the estimated parameters do not match the real ones and drive performance is affected. In the above results, the value of the

Table 7.13: Experimental ranking with different discrete-time models

Indices	Forward Euler	Matrix Fact.	Taylor	Taylor Modified
THD_{is} (%)	4	1	3	2
THD_{T} (%)	4	2	3	1
NRSMD_{Ψ_s} (%)	2	3	4	1
NRSMD_{T} (%)	4	2	3	1
$\tilde{f}_{sw}$ (kHz)	3	2	4	1
Average rank	3.4	2	3.4	$\boxed{1.2}$

equivalent inductance of the machine (σL_s) has been heuristically retuned from the originally estimated value of 11.98 to 14.78 (mH). This change is performed only in the stator current equation, in fact, this parameter affects the torque rate under saturation mode. Finally, experimental results can be improved if a more accurate model of the mechanical subsystem is provided. In fact, viscous and stator friction must be included in the used model.

7.4.2 Weighting Factor Discussion

In PTC, weighting factor of the cost function presented in 4.29 is the parameter to adjust. A starting point to the weighting factor is given by

$$k_{\Psi} = \left(\lambda \frac{T_n}{||\mathbf{\Psi_{s}}_n||}\right)^2, \tag{7.1}$$

where T_n and $||\mathbf{\Psi_{s}}_n||$ are nominal values of torque and stator flux, respectively. The term λ is currently obtained experimentally by a heuristic procedure or running offline simulations. If the last alternative is selected, calculation of an optimal weight factor λ_{opt} is needed, e.g., $\lambda_{opt} = 2.56$ gives the better conditions for THD of stator current (THD_{is}) and torque (THD_T).

Now, if an experimental sweep is developed, the optimal λ is calculated by using THD_{is} and THD_T. Fig. 7.17b shows a parameter sweep from $\lambda = 1.1$ to $\lambda = 5.0$, where Fig. 7.17a illustrates the variation of stator current THD and torque THD with respect to λ. Furthermore, Fig. 7.17c shows the variation of average switching frequency $\tilde{f}_{sw}$ and stator current THD with respect to λ. Finally, with $\lambda_{opt} = 2.56$ is achieved a good trade-off between THD of stator current (THD_{is}) and torque (THD_T). Finally, this value $k_{\Psi} = 4096$ is the experimental optimal weighting factor used in PTC and it is the same obtained in simulation results. The above figure has been constructed using 80 different values of λ. These data were taken at different times of operation of the machine (with a resolution of $\Delta\lambda = 0.05$).

Note that, weighting factor should be tuned online for a high-performance operation through a wide operating range, resulting in a complex drive commissioning [34]. Another solution is presented in [41], where weighting factor is calculated online in an analytical way. This alternative is strongly dependent on the system parameters and requires a comprehensive mathematical analysis.

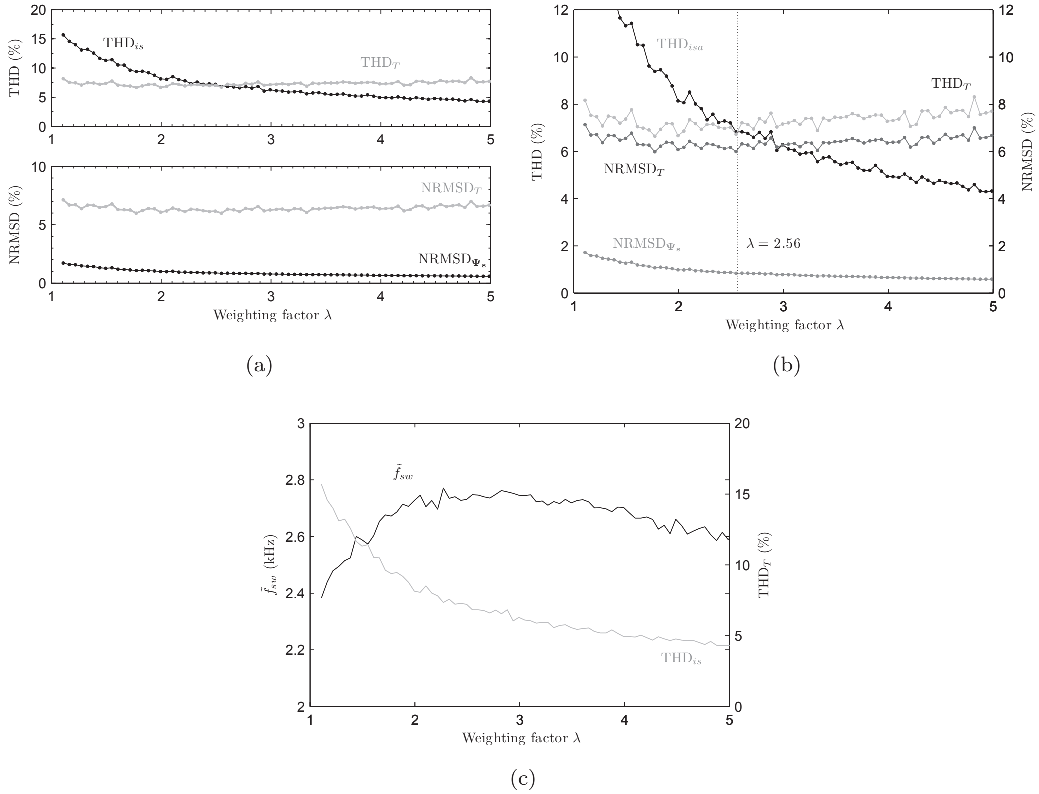


Figure 7.17: Experimental sweep of λ in PTC, (a) variation of THD (top) and NRSMD (bottom) performance indices; (b) selected λ; (c) average switching frequency and stator current THD.

A comparison in steady state between nominal weighting factor used in PTC ($\lambda = 1.0$ or $k_\Psi = 625$) and in PCC ($k_d = 1.0$) is presented in Fig 7.18a and Fig 7.18b respectively. Finally, in applications where the cost function is composed of variables with the same nature (same units and order of magnitude) or it is a decomposition of a single variable into two components, weighting factor tuning is not necessary [45]. Therefore, conventional PCC is recommended instead of PTC for drive applications.

Fig. 7.19 shows the same speed-reversal maneuver presented before but using the conventional schemes with a nominal weighting factor for each case. In PTC, the results are completely different by using an optimal weighting factor; however, in PCC the results are the same with unitary and optimized weighting factor (see Table 7.14).

7.4.3 Ranking Discussion

On the other hand, the conventional method based on weighting factors is a range-dependent because it depends of error differences in each sampling time. With this method the user can give more importance to one objective with respect to another. The importance is included using a weighting factor, in function of the error ranges and the system operation

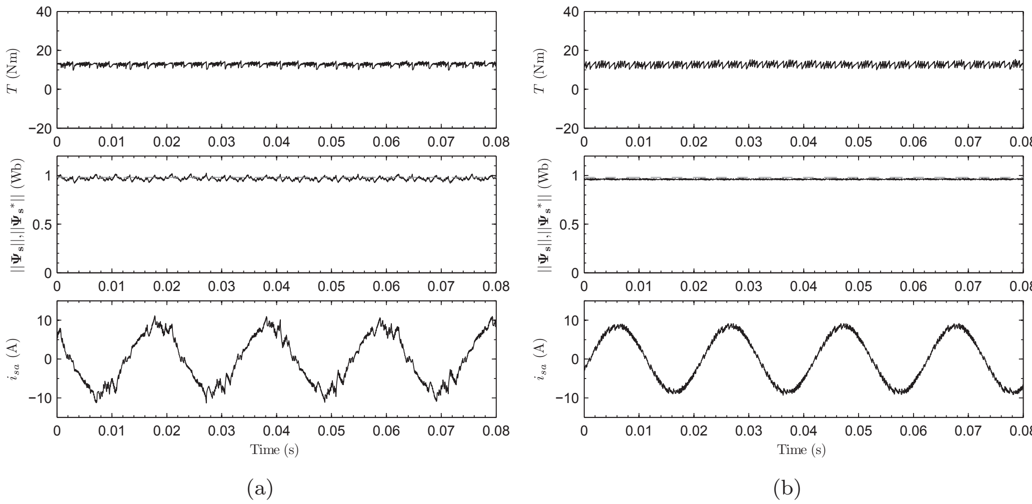


Figure 7.18: Experimental results using the: (a) PTC scheme with $\lambda = 1.0$ ($k_\Psi = 625$); (b) PCC scheme with $k_d = 1.0$, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

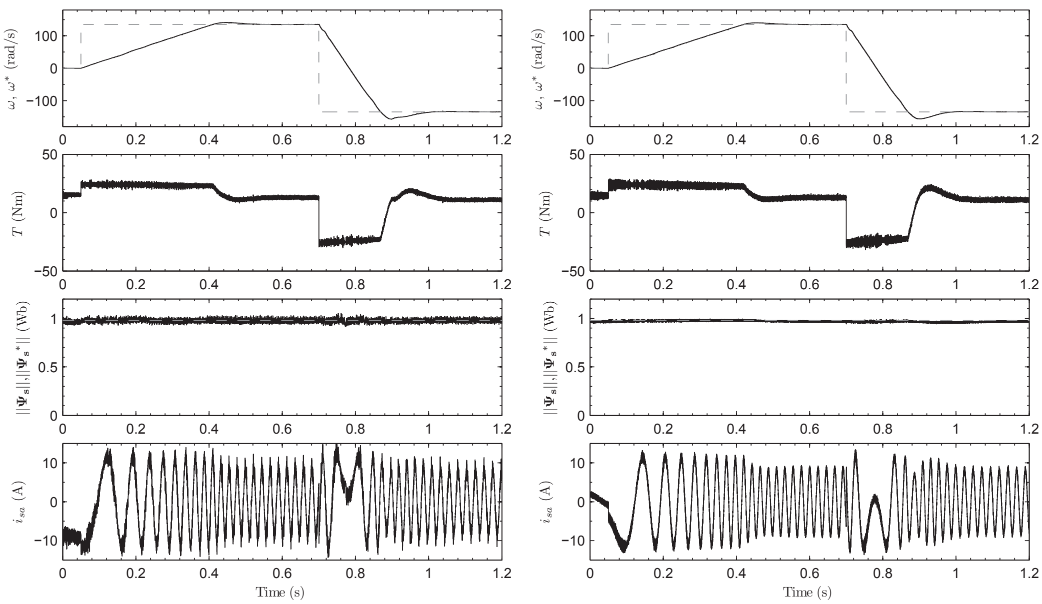
Table 7.14: Indices comparison between conventional schemes

Indices	PTC $\lambda = 1$ ($k_\Psi = 625$)	PTC $\lambda = 2.56$ ($k_\Psi = 4096$)	PCC $k_d = 1$	PCC $k_d = 0.61$
THD_{is} (%)	15.65	6.731	3.993	3.996
THD_T (%)	7.190	7.119	8.922	8.493
NRSMD_{Ψ_s} (%)	1.715	0.836	0.488	0.510
NRSMD_T (%)	6.675	6.634	8.626	8.353
$\tilde{f}_{sw}$ (kHz)	2.412	2.813	2.561	2.632

point. This is an advantage of the weighted approach, but the weighting factors selection is a non-trivial process, specially when more than two objectives are considered in the cost function [43].

The ranking method is a range-independent method, due to it does not depend on differences between error values, transforming the numerical problem (from errors of the objective) to an ordinal problem using rankings. This is a disadvantage, because it considers that objectives can be achieved with the same way; however, for more complex multi-objective systems it is a good solution.

From experimental results, if more weight is assigned to the flux using the conventional PTC, the results will be similar to those obtained with the ranking approach. However, the proposed scheme considers that the objectives should be fairly optimized by means of the transformation from the numerical problem (resulting from errors in the control objectives) to an ordinal problem. This is a disadvantage because the relative difference between the



(a) (b)

Figure 7.19: Experimental results using the: (a) PTC scheme with $\lambda = 1.0$; (b) PCC scheme with $k_d = 1.0$, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 50(%) of the nominal load.

variables is lost, but it is a good solution for complex multiobjective systems because the weighting factor selection is avoided [51].

By using the ranking approach, it is possible for multiple voltage vectors to have the same averaged ranking. To solve this issue, priorities can be assigned for each objective but only for the condition of multiple optimal voltage vectors. For example, the algorithm could select the vector that minimizes the torque error in addition to the average ranking. However, it remains an open question in the proposed algorithm.

Fig. 7.8a shows the speed-reversal maneuver using the PTC scheme with an offline optimized weighting factor. The results are very similar to the ones obtained through the multiobjective ranking approach; however, in the latter method, no additional offline optimization or weighting factor calculation is required.

A comparative table with the main values obtained with PTC and MPTC is presented in Table 7.15. Table 7.15 shows that using the proposed MPTC is achieved a reduction of 1.67 (%), 0.38 (%) and 0.51 (%) in THD_{is}, NRSMD_{Ψ_s} and $\tilde{f}_{sw}$, respectively. However, THD_T and NRSMD_T are increased in 4.53 (%) and 2.88 (%), respectively. In fact, these results are obtained without any weighting factor selection or *a priori* consideration of them, given rise the main advantage of the proposed alternative.

Table 7.15: Summary of experimental results

Indices	Standard PTC	Proposed MPTC	Proposed FPTC
THD_{is} (%)	6.731	5.055	4.264
THD_{T} (%)	7.119	11.65	10.27
NRSMD_{Ψ_s} (%)	0.836	0.454	0.448
NRSMD_{T} (%)	6.634	9.520	8.491
$\tilde{f}_{sw}$ (kHz)	2.813	2.300	2.242
Average calculation time (μs)	11.32	13.16	10.79

7.4.4 Fuzzy Discussion

Calculation process of proposed FPTC could be divided in four sequential but internally parallel stages: cost function calculation, level determination, membership function evaluation, and decision. As previously noted, the internally parallel stages are not required to be evaluated in sequential form. If they are implemented in parallel, an increase of algorithm throughput is possible. However, parallel implementations depend on the choice of implementation technology. If the algorithm is implemented using a conventional DSP-based system, these operations need to be done sequentially. However, if the implementation considers the use of low-cost field programmable gate array (FPGA)-based systems, there is an opportunity to design specialized hardware to perform these tasks in parallel. FPGAs have already been used for power electronics control with excellent results, and even FCS-MPC algorithms have been implemented [44].

If only a DSP is used to implement the algorithm, then the first and second stages could be merged into one. The calculation time of this stage should not introduce significant time with respect to the standard PTC scheme. This is because the number of cost function evaluations is essentially the same as in standard PTC. Two additional comparisons per cost function and voltage vector are added to determine the maximum and minimum levels.

The greatest difference is in the third stage, where membership functions are calculated, which does not exist in the standard scheme. Each cost function and vector evaluation requires a multiplication, which has greater computational cost than additions and comparisons. This is the major factor for the increase of calculation time in algorithm implementation on a DSP. Note that these multiplications are independent and could be implemented in parallel if proper computational resources are available, as in an FPGA-based system.

On the other hand, the proposed method based on FDM is another range-independent method, due to it does not depend of the difference between the error values, transforming the numerical problem (from errors of the objective) to fuzzy sets. However, the conventional method based on weighting factors is a range-dependent because it depends on error differences in each sampling time interval and the user can give more importance to one objective with respect to another. The relative importance is included by using a weighting factor, but they do not mean priorities. Hence, if the optimization process cannot be completed for all objective functions, the method does not indicate, in which sequence objective functions may be discarded [49, 50]. To explain better this dependance, an

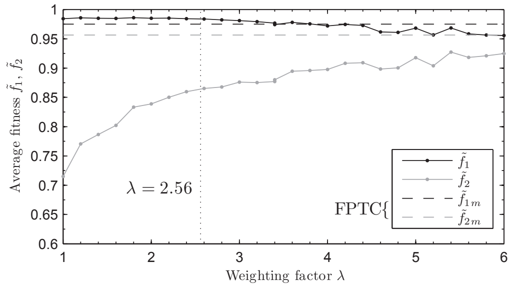


Figure 7.20: Average fitness of torque $\tilde{f}_1$ and stator flux $\tilde{f}_2$ objectives using conventional PTC (solid lines) and FPTC (dash lines).

Table 7.16: Experimental fitness values

Indices	Standard PTC*	Proposed MPTC	Proposed FPTC
Average fitness of torque, $\tilde{f}_1$	0.9839	0.9762	0.9749
Average fitness of stator flux, $\tilde{f}_2$	0.8646	0.9562	0.9564

*with $\lambda = 2.56$.

expression is needed an expression to quantify the achievement of one objective. It can be introduced with an evaluation of the membership functions. These functions represent the degree of attainment or fitness of the goals for each solution [51], thus a fitness of one objective is derived as

$$f(\mathbf{v_s}^{opt}) = \frac{g^{\max} - g(\mathbf{v_s}^{opt})}{g^{\max} - g^{\min}}, \tag{7.2}$$

where, $f(\mathbf{v_s}^{opt})$ and $g(\mathbf{v_s}^{opt})$ are the fitness and numerical error of one objective evaluated in an optimal voltage vector, respectively. The terms $g^{\max}$ and $g^{\min}$ are the maximum and minimum error value for every possible voltage vector, respectively. The fitness value is different in every sampling time for both objectives and depends on the weighting factor [85, 86]. However, is possible to calculate an average fitness value. Fig. 7.20 illustrates the variation of the average fitness value using the conventional PTC in function of the weighting factor λ for both objectives: torque $\tilde{f}_1$ and stator flux $\tilde{f}_2$. Now, with the proposed multiobjective approaches average fitness value is constant and relatively equal for both objectives, Table 7.16.

From Fig. 7.20, by using an optimal weighting factor ($\lambda = 2.56$), the average fitness value of FPTC approach for the flux is higher than using conventional PTC. This point is validated from experimental results, because the ranking approach assigns more importance to the flux objective in comparison with the conventional PTC approach. Furthermore, if more weight is assigned to the flux using the conventional approach ($\lambda \approx 6$), results will be similar with the proposed FPTC approach.

Table 7.17: Experimental indices of different decision functions in FPTC

Indices	`MIN` Operator	`AND` Operator	`AND` Operator*
THD_{is} (%)	4.264	4.564	7.444
THD_{T} (%)	10.27	11.58	7.246
NRSMD_{Ψ_s} (%)	0.448	0.442	0.944
NRSMD_{T} (%)	8.491	9.023	6.542
$\tilde{f}_{sw}$ (kHz)	2.242	2.192	2.066

*with different priorities.

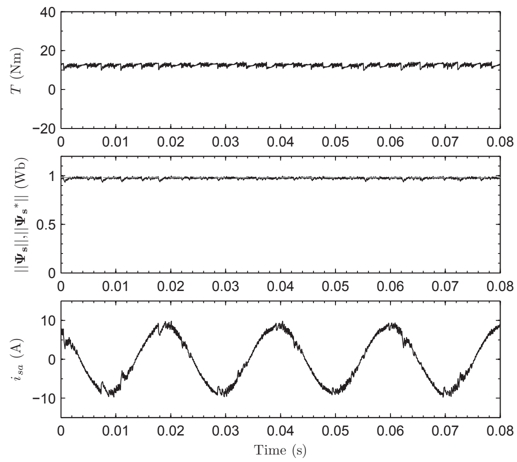


Figure 7.21: Experimental results using the FPTC scheme with different control priorities in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 50 (%) of the nominal load.

Finally, Table 7.15 shows that using the proposed FPTC is achieved a reduction of 0.8 (%) in THD_{is} is achieved with respect to MPTC. Furthermore, these results are obtained without any weighting factor selection. The above results are obtained using a `MIN` operator in the decision function. Similar index values are obtained with `AND` operator, they are presented in Table 7.17. An advantage of FPTC is the possibility to assign different priorities for each control objectives. This is illustrated with an experimental example. In this test, three different control objectives are included according to (5.14). The new control objective is the minimization of quadratic commutation number defined in (4.36). Then, each membership function can be defined with a priority weight k_i, where $\mathbf{k} = [k_1\ k_2\ k_3] = [0.855\ 0.049\ 0.096]$ are the priority weights as defined in (5.14). Fig. 7.21 shows the steady state results of the machine operation with FPTC and the above priorities. These values are selected according to average switching frequency ($\tilde{f}_{sw} \approx 2.0$ (kHz)). Notice that, in this case $||\mathbf{k}||_1 = 1$ and stator current THD is high compared with above cases, due to the priority assigned to stator flux is relatively lower to torque and commutation minimization. Furthermore, the priority

Table 7.18: Algorithm execution times with T_s = 40 (μs). $\bar{x}_t$: Mean value in (μs). σ_t: Standard deviation (μs). t_{max}: Maximum value in (μs).

Time	PTC*			MPTC			FPTC		
	$\bar{x}_t$ (μs)	σ_t (μs)	t_{max} (μs)	$\bar{x}_t$ (μs)	σ_t (μs)	t_{max} (μs)	$\bar{x}_t$ (μs)	σ_t (μs)	t_{max} (μs)
t_m	2.66	0.047	2.91	2.66	0.045	2.91	2.66	0.046	2.91
t_{est}	1.90	0.086	2.10	1.90	0.086	2.10	1.92	0.087	2.13
t_{pred}	6.62	0.045	6.78	5.26	0.042	5.43	5.33	0.042	5.46
t_{opt}	0.13	0.015	0.18	-	-	-	-	-	-
t_{qsort}	-	-	-	1.27	0.122	1.74	-	-	-
t_{rank}	-	-	-	2.07	0.047	2.16	-	-	-
t_{fuzzy}	-	-	-	-	-	-	0.74	0.024	0.84
t_{dec}	-	-	-	-	-	-	0.15	0.012	0.18
Total	11.32	0.107	11.73	13.16	0.188	13.95	$\boxed{10.79}$	0.103	11.25

*without considering the time of weighting factor tuning.

decrement of stator flux is reflected in a better THD and NRSMD of electric torque. Finally, some performance indices of this condition is reported in Table 7.17.

7.4.5 Spectra Analysis

Experimental currents and torque spectra are presented in Fig. 7.22. The machine is operating in steady state with a nominal output frequency (≈48 (Hz)). In PTC the low harmonic frequencies of stator current are important, where the relevant low frequencies are the 5-th and 7-th harmonics, Fig. 7.22a. However, in PTC these harmonic are reduced, specially in the torque spectrum. Now, the spectra of torque in MPTC and MPCC are similars, Fig. 7.22c an Fig. 7.22d. Note that, the low frequencies of torque are bigger than the obtained for conventional schemes. Finally, the current spectra of FPTC and FPCC are more clean than ranking solutions; however, the torque spectra presents a high low frequencies harmonic. Thus, if the application is required a clean current spectrum, the best solution is FPCC, but if a torque spectrum is needed, the recommended scheme is based on conventional PTC.

7.4.6 Computational Burden

On the other hand, the proposed MPTC algorithm adds considerable computational burden. For this reason, a comparison is presented in terms of timing for all algorithms. Table 7.18 shows processing time of each stage considering 12500 iterations in 0.5(s). The time needed for measurements is $t_m = 2.66$ (μs). The time required for estimations, reference generation, and speed controller is $t_{est} = 1.90$ (μs), while the time needed for predictions and delay compensation is $t_{pred} = 5.26$ (μs). The difference in terms of processing time appears in the optimization stage because the time needed for the proposed ranking approach is $t_{opt} = t_{qsort} + t_{rank} = 3.34$ (μs), where $t_{qsort} = 1.27$ (μs) and $t_{rank} = 2.07$ (μs) are used for the quicksort and ranking minimization, respectively, while the time required by the conventional optimization is only $t_{opt} = 0.13$ (μs). This is the main drawback of the MPTC algorithm: the computational burden is 16(%) larger

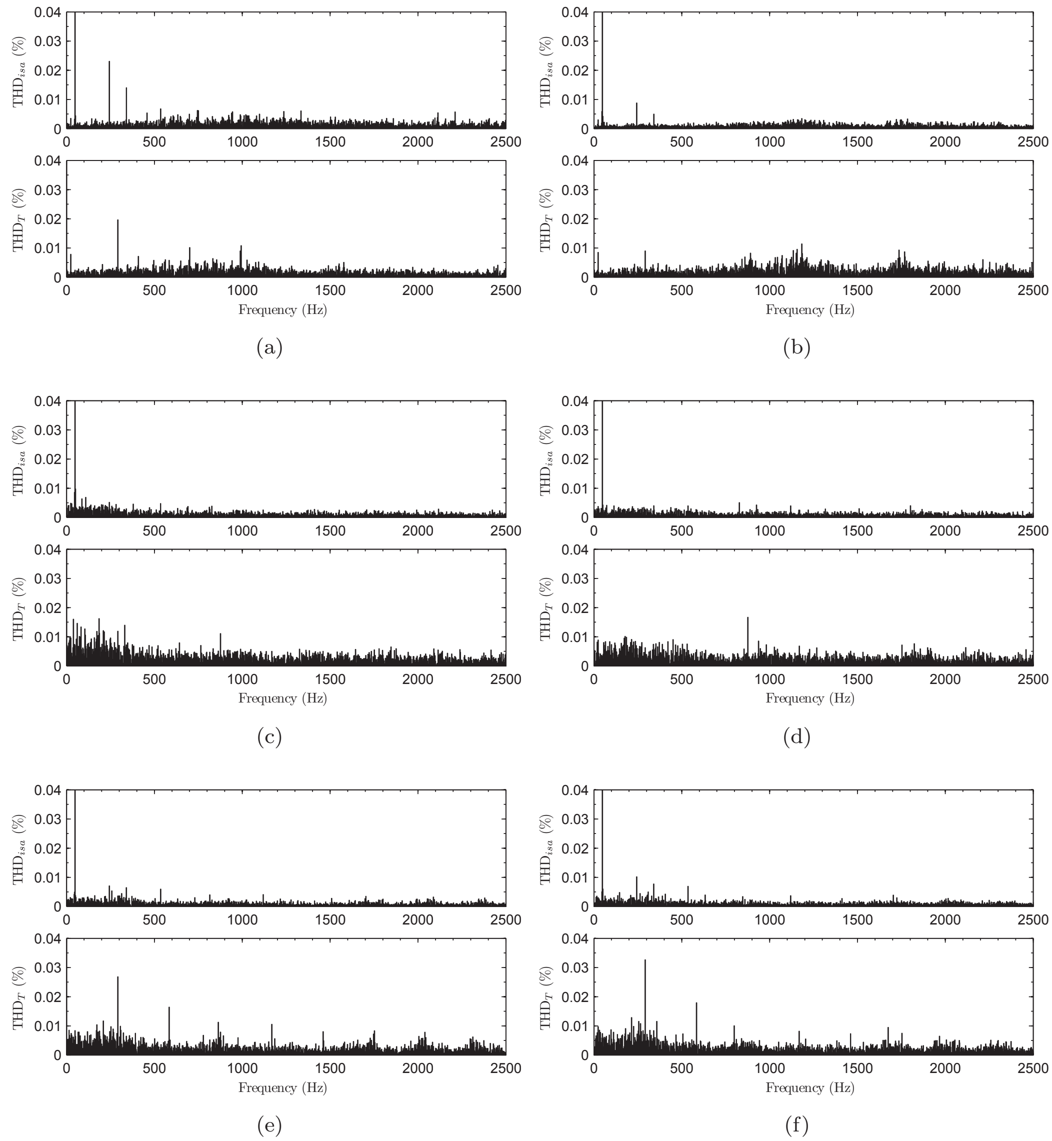


Figure 7.22: Experimental spectra of stator current (top) and torque (bottom) for: (a) PTC; (b) PCC; (c) MPTC; (d) MPCC; (e) FPTC; (f) FPCC, in an induction machine fed by a 2L-VSI. Stator current spectrum (top) and torque spectrum (bottom).

than the standard approach. However, the processing time of the ranking approach depends on the programming, particularly implementation of the quicksort algorithm and ranking optimization [84].

From Table 7.18, it is possible to see that the mean processing time of the standard approach is 11.32 (μs) only. Fortunately, the computational burden of FMPTC is 5(%) lower than the standard approach, because the proposed algorithm is reduced to only two divisions in each sampling time. These operations are used in the membership function identification. This is an important

Table 7.19: Computational burden in terms of voltages vectors n, objectives i and prediction horizons h. Example: 2L-VSI, $n = 8$, $i = 2$, $h = 1$.

Approach	Task	Operations	Example
Conventional Algorithm	Optimization	$(n)^h$	8
	Model	$(n)^h$	8
	Total	$2 \cdot (n)^h$	16
Ranking Algorithm	Quicksort*	$(i \cdot n^2)^h$	128
	Ranking	$(n^i)^h$	64
	Model	$(n)^h$	8
	Total	$(i \cdot n^2)^h + (n^i)^h + (n)^h$	200
Fuzzy Algorithm	Fuzzification	$(3 \cdot i \cdot n)^h$	48
	Decision	$(n)^h$	8
	Model	$(n)^h$	8
	Total	$(3 \cdot i \cdot n^2)^h + 2 \cdot (n)^h$	64

*considering the worst case of comparisons [84].

improvement of the FPTC algorithm, due to with this method the selection of weighting factors is fully avoided. Furthermore, the maximum values for each case are included. This maximum value is an important consideration because the maximum processing time is the minimum sampling time that can be used.

Computational burden grows quickly with the number of voltage vectors, objectives, and prediction horizons. This critical point is introduced in Table 7.19 in terms of operations, such as comparisons and evaluations. For example, using a two-level three-phase inverter, the number of possible voltage vectors is $n = 8$. Then, considering $i = 2$ control objectives and one prediction horizon, the numbers amount of comparisons for the quicksort algorithm and ranking minimization are 128 and 64, respectively, while eight model evaluations are needed. Finally, the total number of operations is around 200. If the number of voltage vectors or prediction horizons is incremented, the number of operations grows quickly. For example, using a three-level three-phase neutral point clamped inverter, the number of possible voltage vectors is $n = 27$. With $i = 3$ control objectives and one prediction horizon, the total number of operations is around 21897. For these implementations, commutation restrictions are needed to reduce the evaluation of the possible voltage vectors [26].

Finally, it is possible to use more complex algorithms based on the ranking and fuzzy approaches to address issues such as priorities, weights, or constraints, such as the switching frequency reduction. However, in the above experimental results, preference was given to maintain the simplicity of the proposed method. For this reason, the main objective of the proposed method only aims to avoid the selection of weighting factors.

7.4.7 Comparison with Field-Oriented Control

In this subsection a brief comparison of one proposed scheme fuzzy decision-making predictive torque control (FPTC) with classical field-oriented control (FOC) scheme is presented. The multiobjective strategy FPTC has been selected due to the experimental performance obtained by comparison with other predictive methods reviewed above. To achieve the fairest possible comparison, some considerations regarding the switching frequency and current sampling have been taken into account. The first one aims to achieve an equivalent switching frequency at least at the specific operation point at which the comparison at steady state is performed. Since FPTC is a variable switching frequency method due to the absence of a modulator, the commutation equivalence between both strategies is achieved by taking the average switching frequency in FPTC as a reference. Then, by modifying the carrier frequency of the modulator in FOC, it is possible to obtain a very similar switching frequency in steady-state.

In a real-time implementation the time required to compute the control law algorithm may take a significant portion of the sample period, resulting in one sampling time delay. This phenomenon is well understood in FOC and may be compensated (see, e.g., [87]). In FPTC algorithm the effect of the time delay has a large impact on the prediction, especially when an one-sample horizon algorithm is considered, and therefore a delay compensation scheme must be implemented [34]. In this comparison the time delay is compensated only in FPTC, but *dc*-link variations are compensated in both schemes.

The operation point chosen is the nominal speed $\omega = 150$ (rad/s) (1440 (rpm)) with a load of 22 (Nm), which corresponds to 88 (%) the nominal torque. The resulting switching frequency of the FPTC strategy is approximately 2.5 (kHz) for each IGBT. Hence, carrier frequency in FOC has been set to 2.5 (kHz), while in FPTC the sampling frequency was set to 25 (kHz).

It is well known that a FOC strategy should be implemented considering synchronized sampling with the peak value of the modulator carrier signal. Thus, the switching ripple of the stator currents is avoided and less noise is introduced into the control loop. For this reason, measured and calculated variables, such as torque or flux, appear without ripple. On the other hand, because FPTC does not need a modulator, a higher sampling frequency is usually required to achieve proper control of the stator current trajectory. The experimental results in FOC have been re-sampled and displayed at the same frequency of FPTC, i.e., 25 (kHz), in such a way that a fair visualization of the switching ripple of the variables in both methods is achieved. In this case is used an approach based on an interrupt triggered by the carrier signal modulation (three-phase PWM) but displayed with an external timer (Timer A). However, as a consequence of the re-sampling, some commutation noise will be introduced and observed in the FOC results. Nevertheless, it can be safely ignored for comparative purposes. Finally, the same external speed PI-controller has been used in FOC and PTC. Furthermore, this control is implemented at a sub-sampled rate (500 (Hz)) to reduce the quantization error in the speed signal derived from an incremental encoder.

The first test shows the performance of both strategies in steady state. The chosen operation point considers a rotor speed $\omega = 150$ (rad/s) with a load $T_l = 22$ (Nm). It is important to highlight that under this condition the average switching frequency for both techniques is equivalent and equal to 2.5 (kHz) for each IGBT. In Fig. 7.23a, the behavior of the resulting stator currents is shown. It can be observed that in FOC the stator current presents a typical PWM waveform with very low distortion. Note that, if a dead-time compensation is performed in FOC, the obtained distortion can be reduced significatively. The current in FPTC has also low distortion, although its ripple is slightly lower compared to FOC, Fig. 7.23b.

To evaluate the distortion of both stator currents, the THD (up to the 50-th harmonic) has been calculated and given in Table 7.20. Spectra of both schemes are presented in Fig. 7.24. It is well known that a FOC strategy has a concentred spectrum due to its fixed switching frequency. Instead of FOC, the proposed FPTC scheme has a sparse spectrum due to the absence of modulator. In fact, spectra are more clean in FOC, specially in torque. Finally, the stator current THD in FPTC is 1.9 (%) lower than classical FOC, Table 7.20. Now, if the torque spectrum is more relevant, FOC is highly superior to the proposed technique.

The following result describes the torque step response for FOC and FPTC. The applied torque step is equal to 25 (Nm) and it was performed by a sudden change in the speed reference from 0 to 120 (rad/s), and hence saturating the speed PI-controller. Both responses are shown and compared in Fig. 7.25, where it is clear that the torque response of FPTC is faster than FOC, being characterized by approximate settling times of 0.4 (ms) versus 3.1 (ms), respectively. Note that in FOC, both PI-controllers have been standardly designed with an overshoot of 5 (%) and a bandwidth of 100 (Hz) and 10 (Hz) for inner and outer control-loop, respectively. However, in

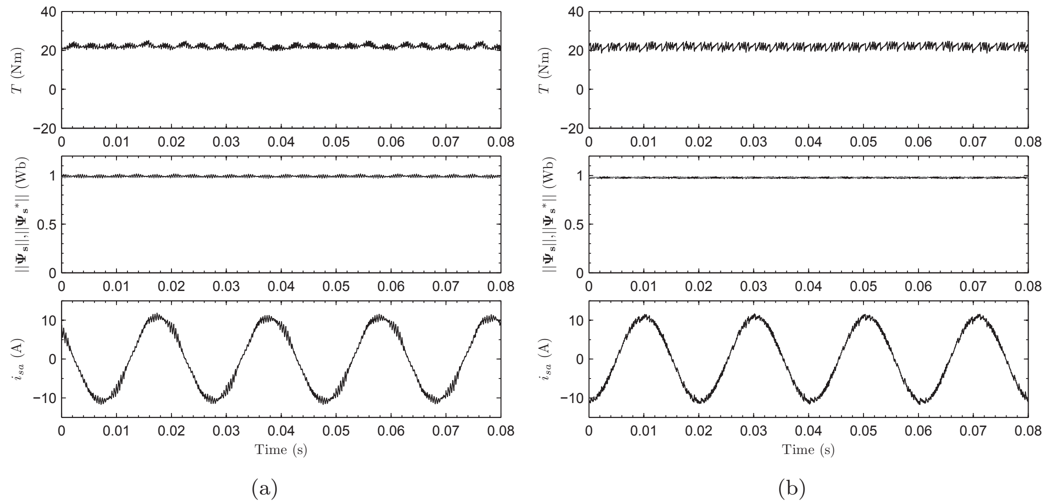


Figure 7.23: Experimental results using the: (a) FOC scheme; (a) FPTC scheme, in an induction machine fed by a 2L-VSI. Stator current, electric torque and stator flux in steady state at 88 (%) of the nominal load.

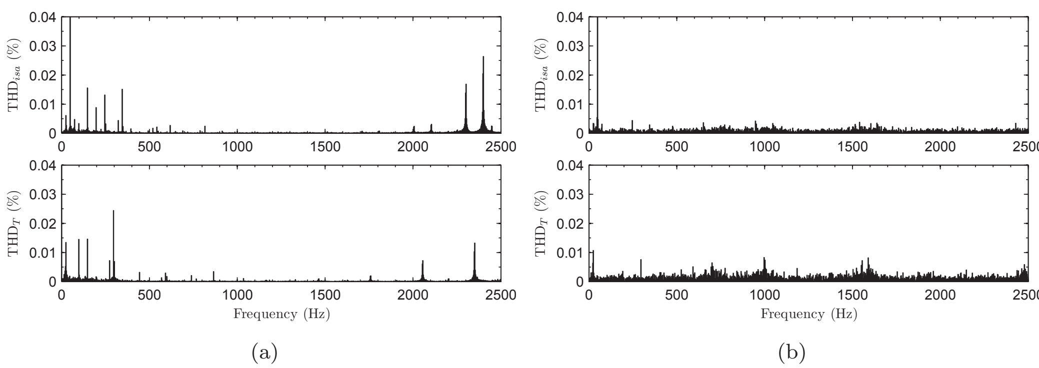


Figure 7.24: Experimental results using the: (a) FOC scheme; (b) FPTC scheme, in an induction machine fed by a 2L-VSI. Stator current spectrum (top) and torque spectrum (bottom).

the literature there are more sophisticated design methods where the dynamic performance of FOC can be significantly improved [1].

The predictive inner loop of FPTC allows to achieve the fastest torque response, limited only by the actuation capability of the inverter, as shown in Fig. 7.25b. In steady state (before and after the torque transient), FPTC alternates the application of active vectors with zero vectors to achieve the appropriate time average of the stator voltage, behaving effectively as kind of a modulator. On the other hand, during the transient, only active vectors are applied, maximizing the actuation and hence minimizing the settling time [34].

A speed reversal maneuver has been performed while the machine is rotating at $\omega = 120$ (rad/s) (80 (%) of nominal speed) and load torque of 22 (Nm) (88 (%) of nominal torque). The results

Table 7.20: Comparative indices between FOC and FPTC

Indices	FOC	FPTC
THD_{is} (%)	5.434	$\boxed{3.593}$
THD_T (%)	4.491	6.041
$NRSMD_{\Psi_s}$ (%)	0.870	0.492
$NRSMD_T$ (%)	4.299	5.657
$\tilde{f}_{sw}$ (kHz)	2.500	2.460
Average calculation time (μs)	5.22	10.79
Settling time (ms)	3.1	0.4

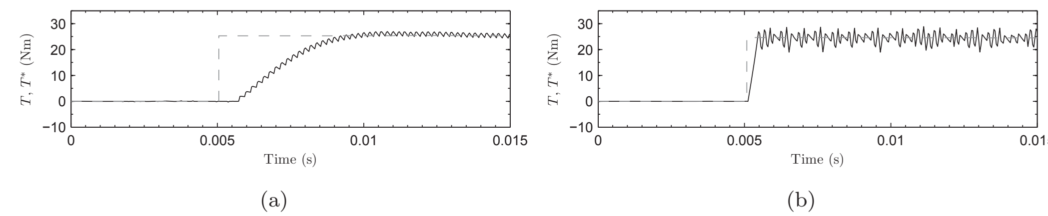


Figure 7.25: Experimental results using the: (a) FOC scheme and (b) FPTC scheme, in an induction machine fed by a 2L-VSI. Torque step response, (gray) torque reference, and (black) torque response.

of speed, torque and stator current of FOC and FPTC are presented in Figs. 7.26a and Figs. 7.26b, respectively. It can be noted that although in both strategies the dynamic performance is appropriate and almost equivalent, at low speed operation, torque ripple in FOC is lower than FPTC.

The main advantage of FOC is its constant switching frequency, resulting in an operation with whistling noise. However, due to direct nature of predictive approaches the switching pattern is random. Thus, there are no peaks in the current spectrum resulting in a spread spectrum, it means that the audible noise of the machine is kind of sizzling noise.

Regarding to the computational effort, the execution time for both strategies has been measured. The results are given in Table 7.20. In fact, FOC time takes half the time of FPTC, it has higher calculation requirements (even when FOC uses one more coordinate transformation), because estimations, predictions and the multiobjective optimization must be computed, for every actuating possibility, in only one sampling step. The above comparison, it is the simplest case, because only seven different vectors are evaluated. Finally, a summary of the different features compared in this work is shown in Table 7.21.

7.5 Conclusions

The specification of weighting factors is a very complex task in implementation of conventional predictive torque control. Weighting factor tuning is replaced by a multiobjective approach, and weighting factor calculation is avoided. In this chapter, multiobjective optimizations has been experimentally validated. The methods are based on the idea that the selected voltage vector should allow a fair minimization of all the objective functions, in this case torque and stator flux

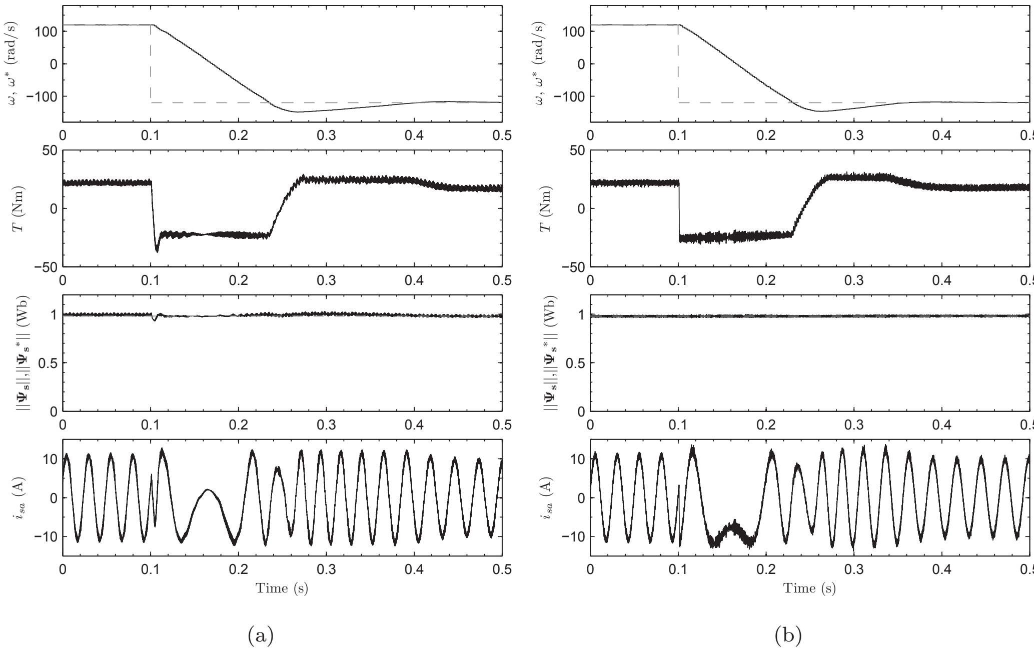


Figure 7.26: Experimental results using the: (a) FOC scheme and (b) FPTC scheme, in an induction machine fed by a 2L-VSI. Speed, electric torque, stator flux, and stator current behavior during a speed-reversal maneuver at 88(%) of the nominal load.

Table 7.21: Comparative issues between strategies.

Feature	FOC	PTC	MPTC	FPTC
Use of Pulse Width Modulator	Yes	No	No	No
PI-current controllers	Yes	No	No	No
Dead-time compensation	Needed	No	No	No
Time-delay compensation	No	Needed	Needed	Needed
dc-link compensation	Yes	Needed	Needed	Needed
Sampling rate	Lower	Higher	Higher	Higher
Switching frequency	Fixed	Variable	Variable	Variable
Conceptual complexity	Lower	Lower	Higher	Higher
Computational cost	Lower	Higher	Higher	Higher
Use of Weighting factor	No	Yes	No	No

tracking.

Two new alternatives to voltage vector selection in the PTC scheme have been presented and applied successfully to the control of an induction motor drive. The first solution is based on a multi-objective ranking-based approach, where the idea is an independent evaluation of each objective function for the converter voltage vectors, and then calculating a ranking of each possible solution using a sorting algorithm. The second proposed strategy uses a fuzzy multicriteria decision-making strategy instead of a conventional aggregate cost function. Through the use of this strategy, the original problem of the determination of the weighting factors is transformed to the design of a suitable decision strategy. The proposed strategy is suitable for realtime applications and it has two degrees of freedom in the form presented in this work: the type of the membership functions and decision function. As evaluated in this work, by using linear membership functions with the conventional minimum operator it is possible to obtain adequate performance. However, there is additional scope for development concerning these functions. From the obtained experimental results in the inverter, the strategy compares favorably with respect to the conventional PTC approach.

From experimental results, good performance in steady state and dynamic behavior were obtained with the proposed strategies, without any offline optimizations such as is done in the standard PTC scheme. In fact, decoupled and fast control of electric torque and stator flux in an induction motor drive is achieved with the multiobjective predictive torque control. Finally, an easy drive commissioning for torque control applications is possible and the process of weighting factor selection is fully avoided.

Chapter 8

CONCLUSIONS

8.1 Summary

BASICALLY two control schemes for electrical drives have dominated industrial applications during the last decades: field-oriented control and direct torque control. Nowadays, these control strategies are fully implemented on digital platforms. In fact, digital signal processors allows high flexibility, the integration of more functionality, and the implementation of more complex control schemes. Due to the development of powerful and fast microprocessors, increasing attention has been dedicated to the use of model predictive control in power electronics. The main approach is based on the calculation of the future system behavior to compute optimal actuation variables.

The research done up to now has revealed that a key issue in FCS-MPC implementations is the selection of the weighting factors used in the cost function. These weighting factors are used to give more importance to one or another variable and to normalize the different control objectives. These scalar factors are parameters to adjust, and its selection is an important task because it is more complex than the tuning of PI coefficients or hysteresis bands of traditional controllers. Several methods using offline and online search procedures have been implemented, but they are strongly dependent on the operation-point and system-parameter. Finally, when more objectives are considered, the weighting factors are usually obtained using heuristic procedures and running time-consuming simulations.

In conventional PTC, the weighting factors associated to torque and stator flux can be reduced to only one scalar factor. In this research this value is adjusted with an offline parameter sweep procedure by using some merit functions, such as total harmonic distortion and normalized root-mean-square deviation. Finally, the weighting factor is selected as the best trade off between stator current and torque THD. The selected value has been evaluated in simulation and validated experimentally. Steady-state and dynamic experimental results shows good speed, flux and torque tracking, while a decoupled stator flux and torque control is successfully developed. Finally, the drive commissioning and effectiveness performance of PTC is strongly dependent on the machine model accuracy and the weighting factor in the objective function. One simple alternative is proposed, where the torque and stator flux control problem is transformed using predictive field-oriented control, where torque and stator flux are commanded by the direct and quadrature component of stator current, respectively. The results of PCC shows that weighting factors are not needed, simplifying the drive commissioning.

Finally, PTC and PCC used a single cost function to solve the optimization problem at each sampling time, but it is not the only possible alternative. The possibility of the use of a different optimizer is proposed in this research. Two different multiobjective optimization methods in order to eliminate the requirement of weighting factors in the predictive torque and flux control scheme are presented. The first optimization problem is solved using a multiobjective ranking approach, giving rise to a multiobjective ranking-based predictive torque and flux control (MPTC). The second approach is based on fuzzy decision-making predictive torque and flux control FPTC. Both multiobjective schemes have been evaluated in simulation and validated experimentally in an induction machine drive fed by a two-level voltage source inverter. Experimental results show the correct steady-state and dynamic operation, while an easy drive commissioning for torque control applications is possible and the process of weighting factor selection is avoided.

8.2 Conclusions

The increasing attention given to FCS-MPC in Power Electronics and drives is remarkable, as reflected in its implementation in a wide range of power topologies and applications. This advance has been made possible in great part by the availability and flexibility of modern digital control platforms, whose ever-increasing computing power is making possible the research of more sophisticated predictive techniques. Several works reported in the recent literature have demonstrated that predictive schemes are an alternative to the classical control solutions, being generally superior in terms of transient performance.

In general, the performance of PTC is strongly system-parameter-dependent. For this reason, in an induction motor drive application an accurate discrete-time model is needed. Several approximate sampled-data models have been studied. In fact, the presented numerical analysis shows that, the discrete-time model based on matrix factorization provides only an approximation although fairly accurate, instead of an exact discretization as it was presented in the literature. Finally, from the comprehensive numerical analysis, matrix factorization, and Taylor approximation are good alternatives to used in an experimental test bench. However, the best discretization model in terms of numerical error and experimental merit functions is achieved with the modified Taylor method.

The selection of the weighting factor is an difficult task in the implementation of conventional PTC. When more objectives are considered in the total cost function, the weighting factors calculation is usually performed using trial and error procedures and running time-consuming simulations. In case of PTC, with the nominal weighting factor is achieved a poor performance, hence an optimized scalar factor is needed. This kind of optimal weighting factor is found by using merit functions or performance indices. The used scalar factor is selected by using offline simulation and validated with experimental results, resulting in a similar value. Finally, the experimental parameter sweep shows that the weighting factor is operation-point, system-parameters and user dependent.

An alternative to PTC has been reported, in fact, finite control set model predictive field-oriented current control or PCC represents a simplification for weighting factor problem due to the difference between magnitude of torque and stator fluxed is avoided. Finally, PCC can be used to accelerate the drive-commissioning. However, with this alternative the stator flux is more coupled with the torque response. To improve the response, another conventional controller can be included to impose the stator or rotor flux directly. In terms of simulation and experimental results, the obtained stator current THD of PCC is lower than PTC scheme. Finally, although in PTC and PCC, the use of the linear combined objective function to solve the optimization problem at each sampling time is straightforward, and for high performance a set of weighting factors must be *a priori* calculated.

To obtain good experimental results an extensive work in relation to the discrete-time modeling and setting of some parameters has been needed to ensure the correct prediction of the stator current, fluxes and torque. Furthermore, the weighting factor of the cost function has been heuristically adjusted and a feedforward of the *dc*-link voltage has been performed to improve the predictions quality. Although the use of the an aggregate objective function to solve the optimization problem at each sampling time is simple, the *a priori* specification of weighting factors is needed. However, it is not the only possible alternative, in fact, with multiobjective optimizations the weighting factor selection is avoided. Finally, the conventional optimization of PTC based on aggregate objective function can fully changed by multiobjective optimizations.

Two new alternatives to solve the optimization process in the predictive torque scheme have been presented and applied successfully to the control of an induction motor drive. These strategies are based on multiobjective optimization methods, and their main advantage is the elimination of the weighting factors, and accordingly the elimination of the guesswork related with them to obtain adequate performance from the control scheme. The proposed control strategies have been tested through simulation and validated with experimental results, confirming from a practical point of view that the schemes have good control performance. Several dynamic tests have been performed, resulting in a good alternative to conventional approaches. In terms of steady state results, merit functions of ranking-based schemes show that it is slightly better in terms of THD, instead of conventional weighted cost function. Finally, a fast and easy drive commissioning for torque control applications is possible with both proposed multiobjective strategies.

In the first proposed method, the weighting factor tuning is replaced by a multiobjective ranking-based approach, and weighting factor calculation is avoided. The method is based on the idea that the selected voltage vector should allow a fair minimization of all the objective functions. In fact, in multiobjective ranking-based PTC the obtained fitness value of stator flux and torque are similar, while in conventional PTC they values are different and weighting-factor dependent. However, if more weight is assigned to the flux using the conventional approach the fitness value will be similar with the proposed MPTC approach. Finally, the major advantage of this proposed alternative is the full tuning-process elimination of inner control-loop parameters.

The second alternative is based on multiobjective fuzzy decision-making PTC, it allows the design of a voltage actuation selector from a higher level approach, instead of tuning weighting factors as in the standard scheme. The discussion has been limited to an algorithm which selects voltage vectors that optimize the required control objectives to the same degree at each sampling time. Naturally, this imposes a fixed tradeoff to the selection stage. In fact, in FPTC the obtained fitness value of stator flux and torque are equivalent. The proposed strategy has two degrees of freedom. These are the type of the membership functions and the used decision function. As evaluated in this work, by using linear membership functions with the conventional minimum operator it is possible to obtain adequate performance. However, there is additional scope for development concerning these functions. From the obtained experimental results in the inverter, the strategy compares favorably to the conventional PTC approach especially in terms of stator current THD, average switching frequency and computational burden.

The proposed FPTC has been compared experimentally with standard field-oriented control. Results have shown that in steady-state operation and under an equivalent number of commutations, both strategies perform as expected. However, if the THD of the stator currents is calculated, FPTC achieves better results than FOC. In transient conditions, the experimental results have verified that FPTC achieves a faster dynamic response due to the absence of internal current loops. The final result has shown that although the computational effort for both strategies is comparable when a two-level inverter is used, the execution time in FPTC is higher than FOC. This is the main drawback of the algorithm: the computational burden is twice than that of the standard approach.

8.3 Future Work

Many variations can be introduced from the multiobjective optimization presented in this research. In particular, consideration of higher level information in the optimization process and faster algorithms for the sorting and filtering processes. More work is required to refine and develop the a posteriori articulation of preferences in this kind of applications such as importance or control priorities. Another possibility is the study of input/output constraints or merit functions inclusion in the optimization stage. For simplicity, the proposed strategies consider horizon-one predictions only. It may be interesting to evaluate the algorithm with longer horizons.

The experimental results presented in this work are attractive enough to justify additional research work to develop more efficient multiobjective PTC strategies and answer the remaining questions. Some of them are related to the parameter sensitivity of this strategy, limitation of the switching frequency and computational optimization to make a feasible option for high power applications, especially for multilevel converters and applications where the objectives should be achieved in the best way possible simultaneously.

Appendix A

PUBLICATIONS DERIVED FROM THIS RESEARCH

The following publications have been partially or completely derived from the research involved in the development of this thesis project.

ISI Journal Publications

[1] C. A. Rojas, J. Yuz, C. Silva, J. Rodriguez, "Comments on "Predictive Torque Control of Induction Machines Based on State-Space Models,"" IEEE Trans. on Ind. Electron., DOI: 10.1109/TIE.2013.2259783, Early Access, 2014.

[2] C. A. Rojas, J. Rodriguez, F. Villarroel, J. Espinoza, C. A. Silva and M. Trincado, "Predictive Torque and Flux Control Without Weighting Factors," IEEE Trans. on Ind. Electron., vol. 60, no. 2, pp. 681-690, Feb. 2013.

[3] F. Villarroel, J. Espinoza, C. A. Rojas, J. Rodriguez, M. Rivera and Daniel Sbárbaro, "Multiobjective Switching State Selector for Finite States Model Predictive Control based on Fuzzy Decision Making in a Matrix Converter," IEEE Trans. on Ind. Electron., vol. 60, no. 2, pp. 578-588, Feb. 2013.

[4] M. Rivera, C. A. Rojas, J. Rodriguez and J. Espinoza, "Methods of Source Current Reference Generation for Predictive Control in a Direct Matrix Converter," IET Power Electron., vol. 6, no. 5, pp. 894-901, May 2013.

[5] M. Rivera, A. Wilson, C. A. Rojas, J. Rodriguez, J. Espinoza, P. Wheeler and L. Empringham, "A Comparative Assessment of Model Predictive Current Control and Space Vector Modulation in a Direct Matrix Converter," IEEE Trans. on Ind. Electron., vol. 60, no. 2, pp. 589-599, Feb. 2013.

[6] J. Rodriguez, M. P. Kazmierkowski, J. Espinoza, P. Zanchetta, H. Abu-Rub, H. A. Young and C. A. Rojas, "State of the Art of Finite Control Set Model Predictive Control in Power Electronics," IEEE Trans. on Ind. Inf., vol. 9, no. 2, pp. 1003-1016, May. 2013.

[7] M. Rivera, J. Rodriguez, J. Espinoza, T. Friedli, T. Kolar, A. Wilson and C. A. Rojas, "Imposed Sinusoidal Source and Load Currents for an Indirect Matrix Converter," IEEE Trans. on Ind. Electron., vol. 59, no. 9, pp. 3427-3435, Sept. 2012.

[8] M. Rivera, J. Rodriguez, P. Wheeler, C. A. Rojas, A. Wilson, and J. Espinoza, “Control of a matrix converter with imposed sinusoidal source currents,” IEEE Trans. on Ind. Electron., vol. 59, no. 4, pp. 1939-1949, Apr. 2012.

[9] J. Rodriguez, R. Kennel, J. Espinoza, M. Trincado, C. A. Silva, and C. A. Rojas, “High performance control strategies for electrical drives: An experimental assessment,” IEEE Trans. on Ind. Electron., vol. 59, no. 2, pp. 812-820, Feb. 2012.

[10] M. Rivera, J. Rodriguez, B. Wu, J. Espinoza, and C. A. Rojas, “Current control for an indirect matrix converter with filter resonance mitigation,” IEEE Trans. on Ind. Electron., vol. 59, no. 1, pp. 71-79, Jan. 2012.

[11] S. Ahmed, A. Iqbal, H. Abu-Rub, J. Rodriguez, C. A. Rojas, and M. Saleh, “Simple carrier based pwm technique for a three-to-nine phase direct ac-ac converter,” IEEE Trans. on Ind. Electron., vol. 58, no. 11, pp. 5014-5023, Nov. 2011.

[12] M. Rivera, C. A. Rojas, J. Rodriguez, P. Wheeler, B. Wu, and J. Espinoza, “Predictive current control with input filter resonance mitigation for a direct matrix converter,” IEEE Trans. on Power Electron., vol. 26, no. 10, pp. 2794-2803, Oct. 2011.

International Conferences

[1] C. A. Rojas, M. A. Perez, J. Rodriguez, A. Wilson, “Reactive Power Control Using a Carrier-Based Modulation for Cascaded Matrix Converter,” in Proc. IECON 2013 - 39th Annual Conf. IEEE Ind. Electron. Society, Accepted for publication, 2013.

[2] M. A. Perez, C. A. Rojas, J. Rodriguez, and H. Abu-Rub, “A Simple Modulation Scheme for a Three-Phase Direct Matrix Converter,” in Proc. IEEE Int. Ind. Electron. (ISIE) Symp., 2012, pp. 105-110.

[3] J. Rodriguez, H. A. Young, C. A. Rojas, S. Kouro, P. Cortés, and and H. Abu-Rub, “Recent Contributions of Predictive Control in Power Electronics and Drives,” in Proc. Workshop on Predictive Control of Electrical Drives and Power Electronics (PRECEDE) 2011, pp. 21-33.

[4] C. A. Rojas, J. Rodriguez, A. Wilson, A. Iqbal, H. Abu-Rub, and S. M. Ahmed, “A Simple Modulation Scheme for a Regenerative Cascaded Matrix Converter,” in Proc. IECON 2011 - 37th Annual Conf. IEEE Ind. Electron. Society, 2011, pp. 4361-4366.

[5] F. Villarroel, J. Espinoza, C. A. Rojas, C. Molina, and J. Rodriguez, “Application of fuzzy decision making to the switching state selection in the predictive control of a direct matrix converter,” in Proc. IECON 2011 - 37th Annual Conf. IEEE Ind. Electron. Society, 2011, pp. 4272-4277.

[6] M. Rivera, J. L. Elizondo, M. E. Macias, O. M. Probst, O. M. Micheloud, J. Rodriguez, C. A. Rojas, and A. Wilson, “Model predictive control of a doubly fed induction generator with an indirect matrix converter,” in Proc. IECON 2010 - 36th Annual Conf. IEEE Ind. Electron. Society, 2010, pp. 2959-2965.

[7] F. Villarroel, J. Espinoza, C. A. Rojas, C. Molina, and E. Espinosa, “A multiobjective ranking based finite states model predictive control scheme applied to a direct matrix converter,” in Proc. IECON 2010 - 36th Annual Conf. IEEE Ind. Electron. Society, 2010, pp. 2941-2946.

[8] J. Rodriguez, B. Wu, M. Rivera, C. A. Rojas, V. Yaramasu, and A. Wilson, “Predictive current control of three-phase two-level four-leg inverter,” in Proc. 14th Int. Power Electron. and Motion Control Conf. (EPE/PEMC), 2010, pp. T3-106-T3-110.

[9] J. Rodriguez, B. Wu, M. Rivera, A. Wilson, V. Yaramasu, and C. A. Rojas, “Model predictive control of three-phase four-leg neutral-point-clamped inverters,” in Proc. Int. Power Electron. Conf. (IPEC), 2010, pp. 3112-3116.

[10] J. Rodriguez, J. Kolar, J. Espinoza, M. Rivera, and C. A. Rojas, "Predictive torque and flux control of an induction machine fed by an indirect matrix converter with reactive power minimization," in Proc. IEEE Int. Ind. Electron. (ISIE) Symp., 2010, pp. 3177-3183.

[11] C. A. Rojas, M. Rivera, J. Rodriguez, A. Wilson, J. Espinoza, F. Villarroel, and P. Wheeler, "Predictive control of a direct matrix converter operating under an unbalanced ac source," in Proc. IEEE Int. Ind. Electron. (ISIE) Symp., 2010, pp. 3159-3164.

[12] V. Yaramasu, J. Rodriguez, B. Wu, M. Rivera, A. Wilson, and C. A. Rojas, "A simple and effective solution for superior performance in two-level four-leg voltage source inverters: Predictive voltage control," in Proc. IEEE Int. Ind. Electron. (ISIE) Symp., 2010, pp. 3127-3132.

[13] J. Rodriguez, J. Espinoza, M. Rivera, F. Villarroel, and C. A. Rojas, "Predictive control of source and load currents in a direct matrix converter," in Proc. IEEE Int. Ind. Tech. (ICIT) Conf., 2010, pp. 1826-1831.

[14] J. Rodriguez, J. Kolar, J. Espinoza, M. Rivera, and C. A. Rojas, "Predictive current control with reactive power minimization in an indirect matrix converter," in Proc. IEEE Int. Ind. Tech. (ICIT) Conf., 2010, pp. 1839-1844.

[15] J. Rodriguez, J. Kolar, J. Espinoza, M. Rivera, and C. A. Rojas, "Predictive torque and flux control of an induction machine fed by an indirect matrix converter," in Proc. IEEE Int. Ind. Tech. (ICIT) Conf., 2010, pp. 1857-1863.

[16] R. Vargas, J. Rodriguez, C. A. Rojas, and P. Wheeler, "Predictive current control applied to a matrix converter: An assessment with the direct transfer function approach," in Proc. IEEE Int. Ind. Tech. (ICIT) Conf., 2010, pp. 1832-1838.

[17] M. Rivera, P. Correa, J. Rodriguez, I. Lizama, J. Espinoza, and C. A. Rojas, "Predictive control with active damping in a direct matrix converter," in Proc. IEEE Energy Conversion Congress and Exp. ECCE 2009, pp. 3057-3062.

Projects related with the research

[1] FONDECYT Project 1100404 - "High Performance Control of Electrical Machine", Powerlab - Universidad Técnica Federico Santa María, Ph.D. José Rodríguez

BIBLIOGRAPHY

[1] M. Kazmierkowski, R. Krishnan, , and F. Blaabjerg, *Control in Power Electronics*. New York: Academic Press, 2002.

[2] N. Mohan, T. Underland, and W. Robbins, *Power Electronics*, 3rd ed. New York: Wiley, 2003.

[3] F. Blaschke, "The principle of field orientation as applied to the new TRANSVECTOR closed-loop control system for rotating machines," *Siemens Review*, vol. 39, no. 5, pp. 217–220, 1972.

[4] K. Hasse, *Control of Electrical Drives*. Berlin: Springer-Verlag, 1996.

[5] W. Leonhard, *Control of Electrical Drives*. Berlin: Springer-Verlag, 1996.

[6] J. Holtz, "Pulse width modulation for electronic power conversion," *Proc. IEEE*, vol. 82, no. 8, pp. 1194–1214, Aug. 1994.

[7] I. Takahashi and T. Noguchi, "A new quick-response and high efficiency control strategy for an induction motor," *IEEE Trans. Ind. Appl.*, vol. 22, no. 5, pp. 820–827, Sep./Oct. 1986.

[8] J. Maciejowski, *Predictive Control with Constraints*. New York: Prentice Hall, 2002.

[9] E. Camacho and C. Bordons, *Model Predictive Control*. Berlin: Springer-Verlag, 2007.

[10] J. Rawlings and D. Mayne, *Model Predictive Control, Theory and Design*. Madison: Nob Hill Publishing, 2009.

[11] J. H. Lee, "Model predictive control: Review of the three decades of development," *Int. Journal of Control, Autom., and Syst.*, vol. 9, no. 3, pp. 415–424, 2011.

[12] M. Morari and J. H. Lee, "Model predictive control: past, present and future," *Comput. and Chem. Eng.*, vol. 23, pp. 667–682, 1999.

[13] J. Holtz and S. Stadtfeldt, "A predictive controller for the stator current vector of AC machines fed from a switched voltage source," in *Proc. IEEE Int. Power Electron. Conf. IPEC*, vol. 2, 1983, pp. 1665–1675.

[14] R. Kennel and D. Schöder, "A predictive control strategy for converters," in *IFAC Control in Power Electron. and Electr. Drives*, 1983, pp. 415–422.

[15] P. Cortes, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodriguez, "Predictive Control in Power Electronics and Drives," *IEEE Trans. Ind. Electron.*, vol. 55, no. 12, pp. 4312–4324, 2008.

[16] A. Linder, R. Kanchan, R. Kennel, and P. Stolze, *Model-Based Predictive Control of Electric Drives*. Munich: Cuvillier Verlag Göttingen, 2010.

[17] S. Mariethoz and M. Morari, "Explicit model-predictive control of a PWM inverter with an LCL filter," *IEEE Trans. Ind. Electron.*, vol. 56, no. 2, pp. 389–399, 2009.

[18] J. Rodriguez, J. Pontt, C. A. Silva, P. Correa, P. Lezana, P. Cortes, and U. Ammann, "Predictive current control of a voltage source inverter," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 495–503, 2007.

[19] T. Laczynski and A. Mertens, "Predictive Stator Current Control for Medium Voltage Drives With LC Filters," *IEEE Trans. Power Electron.*, vol. 24, no. 11, pp. 2427–2435, Nov. 2009.

[20] P. Cortes, A. Wilson, S. Kouro, J. Rodriguez, and H. Abu-Rub, "Model Predictive Control of Multilevel Cascaded H-Bridge Inverters," *IEEE Trans. Ind. Electron.*, vol. 57, no. 8, pp. 2691–2699, Aug. 2010.

[21] F. Defay, A.-M. Llor, and M. Fadel, "Direct Control Strategy for a Four-Level Three-Phase Flying-Capacitor Inverter," *IEEE Trans. Ind. Electron.*, vol. 57, no. 7, pp. 2240–2248, Jul. 2010.

[22] S. Muller, U. Ammann, and S. Rees, "New time-discrete modulation scheme for matrix converters," *IEEE Trans. Ind. Electron.*, vol. 52, no. 6, pp. 1607–1615, Dec. 2005.

[23] M. J. Duran, J. Prieto, F. Barrero, and S. Toral, "Predictive Current Control of Dual Three-Phase Drives Using Restrained Search Techniques," *IEEE Trans. Ind. Electron.*, vol. 58, no. 8, pp. 3253–3263, 2011.

[24] F. Morel, X. Lin-Shi, J.-M. Retif, B. Allard, and C. Buttay, "A Comparative Study of Predictive Current Control Schemes for a Permanent-Magnet Synchronous Machine Drive," *IEEE Trans. Ind. Electron.*, vol. 56, no. 7, pp. 2715–2728, 2009.

[25] E. J. Fuentes, A. Silva, and J. I. Yuz, "Predictive Speed Control of a Two-Mass System Driven by a Permanent Magnet Synchronous Motor," *IEEE Trans. Ind. Electron.*, vol. 59, no. 7, pp. 2840–2848, 2012.

[26] M. Preindl, E. Schaltz, and P. Thogersen, "Switching Frequency Reduction Using Model Predictive Direct Current Control for High-Power Voltage Source Inverters," *IEEE Trans. Ind. Electron.*, vol. 58, no. 7, pp. 2826–2835, 2011.

[27] M. Preindl and E. Schaltz, "Sensorless Model Predictive Direct Current Control Using Novel Second-Order PLL Observer for PMSM Drive Systems," *IEEE Trans. Ind. Electron.*, vol. 58, no. 9, pp. 4087–4095, 2011.

[28] H. Miranda, P. Cortes, J. I. Yuz, and J. Rodriguez, "Predictive torque control of induction machines based on state-space models," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1916–1924, 2009.

[29] J. Martinez, R. Kennel, and T. Geyer, "Model predictive direct current control," in *IEEE Int. Conf. Ind. Technol. ICIT*, Mar. 14-17, 2010, pp. 1808–1813.

[30] T. Geyer, G. Papafotiou, and M. Morari, "Model Predictive Direct Torque Control - Part I: Concept, Algorithm, and Analysis," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1894–1905, Jun. 2009.

[31] J. Rodriguez, M. Kazmierkowski, J. Espinoza, P. Zanchetta, H. Abu-Rub, H. Young, and C. Rojas, "State of the Art of Finite Control Set Model Predictive Control in Power Electronics," *IEEE Trans. Ind. Informat.*, vol. 9, no. 2, pp. 1003–1016, May 2013.

[32] S. Kouro, P. Cortes, R. Vargas, U. Ammann, and J. Rodriguez, "Model predictive control - A simple and powerful method to control power converters," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1826–1838, 2009.

[33] T. Geyer, "A Comparison of Control and Modulation Schemes for Medium-Voltage Drives: Emerging Predictive Control Concepts Versus PWM-Based Schemes," *IEEE Trans. Ind. Appl.*, vol. 47, no. 3, pp. 1380–1389, 2011.

[34] J. Rodriguez, R. M. Kennel, J. R. Espinoza, M. Trincado, C. A. Silva, and C. A. Rojas, "High-performance control strategies for electrical drives: An experimental assessment," *IEEE Trans. Ind. Electron.*, vol. 59, no. 2, pp. 812–820, 2012.

[35] R. Aguilera and D. Quevedo, "On stability and performance of finite control set MPC for power converters," in *Proc. Workshop Predictive Control of Electr. Drives and Power Electron. (PRECEDE)*, 2011, pp. 55–62.

[36] R. Aguilera, “Stability Analysis of Quadratic MPC with a Discrete Input Alphabet,” *IEEE Trans. Ind. Appl.*, Early Access, 2013.

[37] C. Muller, D. Quevedo, and G. Goodwin, “How Good is Quantized Model Predictive Control With Horizon One?” *IEEE Trans. Autom. Control*, vol. 56, no. 11, pp. 2623–2638, 2011.

[38] R. Aguilera, P. Lezana, and D. Quevedo, “Finite-Control-Set Model Predictive Control With Improved Steady-State Performance,” *Industrial Informatics, IEEE Transactions on*, vol. 9, no. 2, pp. 658–667, 2013.

[39] J. Rodriguez, J. Pontt, C. Silva, P. Cortés, U. Ammann, and S. Rees, “Predictive Direct Torque Control of an Induction Machine,” in *Proc. IEEE Int. Power Electron. Motion Control Conf. EPE-PEMC*, Sep. 2-4, 2004.

[40] M. Ehrgott and X. Gandibleux, *Multiple Criteria Optimization: State of the Art Annotated Bibliographic Surveys*. New York: Kluwer Academic, 2003.

[41] S. Davari, D. Arab Khaburi, and K. Ralph, “An improved FCS-MPC Algorithm for Induction Motor With Imposed Optimized Weighting Factor,” *IEEE Trans. Power Electron.*, vol. 27, no. 3, pp. 1540–1551, 2011.

[42] R. Vargas, U. Ammann, B. Hudoffsky, J. Rodriguez, and P. Wheeler, “Predictive torque control of an induction machine fed by a matrix converter with reactive input power control,” *IEEE Trans. Power Electron.*, vol. 25, no. 6, pp. 1426–1438, 2010.

[43] P. Zanchetta, “Heuristic multi-objective optimization for cost function weights selection in finite states model predictive control,” in *Proc. Workshop Predictive Control of Electr. Drives and Power Electron. (PRECEDE)*, 2011, pp. 70–75.

[44] S. Thielemans, T. J. Vyncke, and J. Melkebeek, “Weight factor selection for model-based predictive control of a four-level flying capacitor inverter,” *Power Electron. IET*, vol. 5, no. 3, pp. 323–333, 2012.

[45] P. Cortes, S. Kouro, B. La Rocca, R. Vargas, J. Rodriguez, J. Leon, S. Vazquez, and L. Franquelo, “Guidelines for weighting factors design in model predictive control of power converters and drives,” in *Proc. IEEE Int. Conf. Ind. Technol. ICIT 2009*, 2009, pp. 1–7.

[46] C. A. Rojas, J. Rodriguez, F. Villarroel, J. R. Espinoza, C. A. Silva, and M. Trincado, “Predictive torque and flux control without weighting factors,” *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 681–690, 2013.

[47] F. Villarroel, J. R. Espinoza, C. A. Rojas, J. Rodriguez, M. Rivera, and D. Sbarbaro, “Multiobjective switching state selector for finite-states model predictive control based on fuzzy decision making in a matrix converter,” *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 589–599, 2013.

[48] G. P. Liu, J. B. Yang, and J. F. Whidborne, *Multiobjective Optimisation and Control*. Baldock, Hertfordshire: Research Studies Press Ltd., 2003.

[49] A. Gambier and E. Badreddin, “Multi-objective Optimal Control: An Overview,” in *IEEE Int. Conf. on Control Appl., CCA*, 2007, pp. 170–175.

[50] A. Gambier, “MPC and PID Control Based on Multi-Objective Optimization,” in *Proc. American Control Conf.*, 2008, pp. 4727–4732.

[51] P. J. Bentley and J. P. Wakefield, “Finding Acceptable Solutions in the Pareto-Optimal Range using Multiobjective Genetic Algorithms,” in *Soft Computing in Engineering Design and Manufacturing, Berlin/Heidelberg, Germany*, 1998, pp. 231–240.

[52] L. Zadeh, “Fuzzy Sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[53] T. Geyer, G. Papafotiou, and M. Morari, “Model Predictive Control in Power Electronics: A Hybrid Systems Approach,” in *IEEE Conf. on Decision and Control - European Control Conf., CDC-ECC*, 2005, pp. 5606–5611.

[54] A. Bemporad, F. Borrelli, and M. Morari, "Model predictive control based on linear programming - the explicit solution," *IEEE Trans. Autom. Control*, vol. 47, no. 12, pp. 1974–1985, 2002.

[55] M. Kvasnica, P. Grieder, and M. Baotić, "Multi-Parametric Toolbox (MPT)," 2004. [Online]. Available: http://control.ee.ethz.ch/~mpt/

[56] T. Geyer, "Low Complexity Model Predictive Control in Power Electronics and PowerSystems," Ph.D. dissertation, Swiss Federal Institute of Technology (ETH) Zurich, 2005.

[57] B. Zhang, X. Wang, Y. Tang, and W. Zhang, "Predictive functional control and stability analysis of MIMO bilinear systems," in *American Control Conference*, Jun. 2006, pp. 4854–4858.

[58] C. A. Silva and J. I. Yuz, "On sampled-data models for model predictive control," in *Proc. IEEE Ind. Electron. Soc. Annual Conf., IECON*, 2010, pp. 2966–2971.

[59] Y. Zhang, J. Zhu, and W. Xu, "Analysis of one step delay in direct torque control of permanent magnet synchronous motor and its remedies," in *Proc. Int Electrical Machines and Syst. (ICEMS) Conf.*, 2010, pp. 792–797.

[60] C. Bottura, J. Silvino, and P. de Resende, "A flux observer for induction machines based on a time-variant discrete model," *IEEE Trans. Ind. Appl.*, vol. 29, no. 2, pp. 349–354, 1993.

[61] J. Holtz, "The dynamic representation of AC drive systems by complex signal flow graphs," in *Proc. IEEE Int Ind. Electron. Symp., ISIE*, 1994, pp. 1–6.

[62] B. Joshi and M. Chandorkar, "Time discretization issues in induction machine model solving for real-time applications," in *Proc. IEEE Int. Electric Machines Drives Conf. (IEMDC),*, 2011, pp. 675–680.

[63] L.-J. Diao, D. nan Sun, K. Dong, L.-T. Zhao, and Z.-G. Liu, "Optimized Design of Discrete Traction Induction Motor Model at Low-Switching Frequency," *IEEE Trans. Power Electron.*, vol. 28, no. 10, pp. 4803–4810, 2013.

[64] J. Riveros, F. Barrero, E. Levi, M. Duran, S. Toral, and M. Jones, "Variable-Speed Five-Phase Induction Motor Drive Based on Predictive Torque Control," *IEEE Trans. Ind. Electron.*, vol. 60, no. 8, pp. 2957–2968, 2013.

[65] M. J. Duran, J. A. Riveros, F. Barrero, H. Guzman, and J. Prieto, "Reduction of Common-Mode Voltage in Five-Phase Induction Motor Drives Using Predictive Control Techniques," *IEEE Trans. Ind. Appl.*, vol. 48, no. 6, pp. 2059–2067, 2012.

[66] M. Visser, S. Stramigioli, and C. Heemskerk, "Cayley-Hamilton for roboticists," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2006, pp. 4187–4192.

[67] D. S. Bernstein, *Matrix Mathematics: Theory, Facts, and Formulas with Application to Linear Systems Theory.* Princeton, NJ: Princeton University Press, 2009.

[68] S. GuerMah, S. Djennoune, and M. Bettayeb, "Controllability and Observability of Lineal Discrete-Time Fractional-Order Systems," *Int. J. Appl. Math. Comput. Sci.*, vol. 18, no. 2, pp. 213–222, 2008.

[69] C. H. Houpis and G. B. Lamont, *Digital Control Systems: Theory, Hardware, Software.* New York: McGraw-Hil, 1992.

[70] M. Preindl and S. Bolognani, "Model Predictive Direct Speed Control with Finite Control Set of PMSM Drive Systems," *IEEE Trans. Power Electron.*, vol. 28, no. 2, pp. 1007–1015, 2013.

[71] G. Goddwin, S. Graebe, and M. Salgado, *Control System Design.* Prentice-Hall, 2001.

[72] T. Nussbaumer, M. L. Heldwein, G. Gong, S. D. Round, and J. W. Kolar, "Comparison of prediction techniques to compensate time delays caused by digital control of a three-phase Buck-type PWM rectifier system," *IEEE Trans. Ind. Electron.*, vol. 55, no. 2, pp. 791–799, 2008.

[73] G. Papafotiou, J. Kley, K. G. Papadopoulos, and M. Bohren, P.and Morari, "Model predictive direct torque control - Part II: Implementation and experimental evaluation," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1906–1915, 2009.

[74] Y. Zhang, J. Zhu, and W. Xu, "Predictive torque control of permanent magnet synchronous motor drive with reduced switching frequency," in *Proc. Int Electrical Machines and Syst. (ICEMS) Conf.*, 2010, pp. 798–803.

[75] R. Parreiras, J. Maciel, and J. Vasconcelos, "The A Posteriori Decision in Multiobjective Optimization Problems With Smarts, Promethee II, and a Fuzzy Algorithm," *Magnetics, IEEE Transactions on*, vol. 42, no. 4, pp. 1139–1142, 2006.

[76] Rachmawati, L. and Srinivasan, D., "Preference Incorporation in Multi-objective Evolutionary Algorithms: A Survey," in *IEEE Congr. on Evol. Comput. (CEC)*, 2006, pp. 962–968.

[77] M. Castillo Tapia and C. A. Coello Coello, "Applications of multi-objective evolutionary algorithms in economics and finance: A survey," in *IEEE Congr. on Evol. Comput. (CEC)*, 2007, pp. 532–539.

[78] N. Pindoriya, S. Singh, and K. Lee, "A Comprehensive Survey on Multi-Objective Evolutionary Optimization in Power System Applications," in *IEEE Power and Energy Society General Meeting (PES)*, 2010, pp. 1–8.

[79] P. Y. Ekel, M. Menezes, and F. H. S. Neto, "Decision making in a fuzzy environment and its application to multicriteria power engineering problems," in *Proc. of the Int. Conf. on Hybrid Syst. and Appl.*, vol. 1, no. 4, 2007, pp. 527–536.

[80] J. da Costa Sousa and U. Kaymak, "Model predictive control using fuzzy decision functions," *IEEE Trans. Syst., Man, Cybern. B*, vol. 31, no. 1, pp. 54–65, Feb. 2001.

[81] J. Sousa and U. Kaymak, *Fuzzy decision making in modeling and control*. New Jersey: World Scientific, 2002.

[82] W. Kong, J. Ding, T. Chai, and J. Sun, "Large-dimensional multi-objective evolutionary algorithms based on improved average ranking," in *Proc. 49th IEEE Conf. Decision and Control (CDC)*, 2010, pp. 502–507.

[83] B. W. Kernighan and D. M. Ritchie, *The C Programming Language, 2nd Ed.* New Jersey: Prentice Hall, 1988.

[84] S. S. Skiena, *The Algorithm Design Manual*. London: Springer-Verlag, 2008.

[85] W. A. Lodwick and J. Kacprzyk, *Fuzzy Optimization: Recent Advances and Applications*. Berlin/Heidelberg: Springer-Verlag, 2010.

[86] T. Okabe, Y. Jin, and B. Sendhoff, "A critical survey of performance indices for multi-objective optimisation," in *IEEE Congr. on Evol. Comput. (CEC)*, vol. 2, 2003, pp. 878–885.

[87] B.-H. Bae and S.-K. Sul, "A compensation method for time delay of full-digital synchronous frame current regulator of PWM AC drives," *IEEE Trans. Ind. Appl.*, vol. 39, no. 3, pp. 802–810, May./Jun. 2003.

COPYRIGHT

CURRICULUM VITAE

Name:	Christian Alexis Rojas Monrroy
Birth:	December 13, 1984 - Vallenar, Chile
Nationality:	Chilean

Education:

August 2009 to date	Ph.D in Power Electronics Universidad Técnica Federico Santa María (UTFSM), Valparaíso, Chile
2003 - 2009	Electronic Engineering Universidad de Concepción (UdeC), Concepción, Chile
1998 - 2002	Secondary Education Escuela Técnico Profesional, Fund. U.D.A, Copiapó, Chile

Experience:

March 2013 to date	Instructor Professor, “ELO-314 Digital Signal Processing Laboratory” Department of Electronics - UTFSM
March 2011 - July 2011	Assistant, “IPD-413 Advanced Seminar of Industrial Electronics” Department of Electronics - UTFSM, Ph.D. José Rodríguez
March 2010 - March 2012	Research Assistant on Power Electronics FONDECYT Project 1100404 “High Performance Control of Electrical Machine” Powerlab - UTFSM, Ph.D. José Rodríguez
April 2009 - March 2010	Research Assistant on Power Electronics Powerlab - UTFSM, Ph.D. José Rodríguez
April 2008 - April 2009	Research Assistant on Power Electronics FONDECYT Project 1080059 “Control of Indirect Matrix Converters” LCDA - UdeC, Ph.D. José Espinoza

Awards

2011	IES Student Scholarship, 37^{th} Annual Conf. of the IEEE-IES IECON 2011 Conference, Melbourne, Australia
2010-2011	Incentive Program for Scientific Initiation (PIIC) Powerlab - UTFSM, Ph.D. José Rodríguez
2010 to date	Postgraduate Scholarship to fund the Ph.D studies DGIP - UTFSM
2010 to date	Doctoral scholarship from the Chilean government CONICYT
2009	Engineering Faculty Award, UdeC Graduated with highest distinction of Electronics Engineering, Graduates ranking: 2^{nd} place