COPPE
Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia
UFRJ

A CONTRIBUTION TO MODEL PREDICTIVE CONTROLLERS WITH FIXED SWITCHING FREQUENCY AND LOW COMPUTATIONAL COST

Thiago Cardoso Tricarico

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientador: Mauricio Aredes

Rio de Janeiro
Julho de 2022

A CONTRIBUTION TO MODEL PREDICTIVE CONTROLLERS WITH FIXED SWITCHING FREQUENCY AND LOW COMPUTATIONAL COST

Thiago Cardoso Tricarico

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Orientador: Mauricio Aredes

Aprovada por: Prof. Mauricio Aredes

Prof. Luís Guilherme Barbosa Rolim

Prof. Lucas Frizera Encarnação

Prof. Bruno Wanderley França

Prof. Leandro Michels

RIO DE JANEIRO, RJ – BRASIL
JULHO DE 2022

Tricarico, Thiago Cardoso

A Contribution To Model Predictive Controllers With Fixed Switching Frequency And Low Computational Cost/Thiago Cardoso Tricarico. – Rio de Janeiro: UFRJ/COPPE, 2022.

XIX, 113 p.: il.; 29, 7cm.

Orientador: Mauricio Aredes

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2022.

Referências Bibliográficas: p. 82 – 95.

1. Model Predictive Control. 2. Switching Frequency Spread. 3. Jaya Algorithm. I. Aredes, Mauricio. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

"Dedico esta tese de doutorado à
minha mãe, no seu belo jardim
de margaridas"

Agradecimentos

Aos meus pais e aos meus queridos irmãos que me apoiaram ao longo de toda esta jornada, especialmente à minha mãe, Maria Izabel, que sempre soube dizer as palavras certas quando tive dúvidas.

A minha avó Odette que não poupou esforços ao longo da vida, por mim, pelos meus irmãos e pela qualidade de nossa educação.

Aos meus colegas do Laboratório de Eletrônica de Potência e Média Tensão (LEMT), pela ajuda ao longo do doutorado, em especial ao Marcello e ao Leonardo que foram fundamentais para a obtenção da comprovação experimental desta tese.

Ao meu orientador, Mauricio Aredes, que em uma tarde de janeiro sem nunca ter me conhecido me acolheu em seu laboratório e me deu uma oportunidade que mudou minha vida. Muito obrigado, Mauricio.

Ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), pelo apoio financeiro nesta pesquisa.

A minha amada esposa Danielle, que esteve comigo todos esses anos, compartilhando sonhos e decepções, alegrias e tristezas, sem nunca ter deixado de me amparar e apoiar com seu sorriso.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

UMA CONTRIBUIÇÃO AOS CONTROLADORES PREDITIVOS BASEADOS EM MODELO COM FREQUÊNCIA DE CHAVEAMENTO FIXA E BAIXO CUSTO COMPUTACIONAL

Thiago Cardoso Tricarico

Julho/2022

Orientador: Mauricio Aredes

Programa: Engenharia Elétrica

Esta tese propõe uma nova técnica de controle preditivo baseado em modelo para aplicações em conversores de potência. Esta técnica é baseada em um recente algoritmo meta-heurístico, chamado algoritmo Jaya, como o otimizador do MPC. A solução proposta apresenta três benefícios: implementação simples, custo computacional viável, e frequência de chaveamento fixa. A estratégia proposta de MPC, referida como Jaya-MPC, destaca-se como uma alternativa ao FCS-MPC que apresenta uma frequência de chaveamento variável. Além disso, este trabalho propõe uma contribuição secundária: uma nova métrica, que avalia o espalhamento do perfil da frequência de chaveamento de conversores de potência, avaliando a variabilidade da frequência de chaveamento produzida por controladores preditivos. Este trabalho utiliza esta métrica para investigar o perfil de frequência de chaveamento do FCS-MPC. Um método simples de parametrização do algoritmo de controle proposto surge de uma análise paramétrica, resultando numa solução com alta qualidade de energia. Além disso, este trabalho compara o custo computacional, baseado no número de predições, tanto do controlador proposto como do paradigma do MPC na eletrônica de potência. Os resultados experimentais provam que a estratégia proposta é uma técnica de implementação simples com um custo computacional viável e elevada qualidade de energia.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

A CONTRIBUTION TO MODEL PREDICTIVE CONTROLLERS WITH FIXED SWITCHING FREQUENCY AND LOW COMPUTATIONAL COST

Thiago Cardoso Tricarico

July/2022

Advisor: Mauricio Aredes

Department: Electrical Engineering

This thesis proposes a new Model Predictive Control (MPC) technique for power-converter applications, based on a recent meta-heuristic algorithm, called the Jaya algorithm, as the optimizer of MPC. The proposed solution presents three benefits: simple implementation, viable computational cost, and fixed switching frequency. The proposed MPC strategy, referred to as Jaya-MPC, stands out as an alternative to the classical Finite-Control-Set Model Predictive control (FCS-MPC), which presents a variable switching frequency. In addition, this work proposes a secondary contribution: a new metric, which assesses the spread of the switching frequency profile of power converters, i.e., it evaluates the variability of the switching frequency produced by MPC. This work uses this metric to investigate the switching frequency profile of FCS-MPC. A straightforward parameter-setting method of the proposed algorithm rises from a parametric analysis, resulting in a high-power-quality solution. Also, this work compares the computational cost, based on the number of predictions, of both the proposed controller and the paradigm of MPC in power electronics. Experimental outcomes prove that Jaya-MPC is a simple-implementation technique with viable computational cost and high power quality.

Contents

List of Figures

List of Tables

List of Algorithms

List of Symbols

J	cost function, p. 34
L	filter inductance, p. 40
NPP	number of predictions per period, p. 59
R	filter resistance, p. 40
T_{base}	base period, p. 59
T_o	normalized period, p. 59
T_s	sampling period, p. 40
p	Sampling and fundamental frequencies' ratio, p. 58
$\mathbf{u_b}$	best solution of the population, p. 35
p^{sp}	active power reference, p. 39
q^{sp}	reactive power reference, p. 39
f_n^{sw}	Switching frequency components, p. 53
f_{sw}^{max}	Maximum switching frequency, p. 54
$f_{sw}^{\star}$	Dominant frequency, p. 52
q	Sampling and dominant frequencies' ratio, p. 57
$\bar{f}_{sw}$	Average switching frequency, p. 15
$\hat{f}_{sw}$	Instantaneous switching frequency, p. 52
h_n^{sw}	Switching frequency components' weights, p. 53
$h_{sw}^{\star}$	Weight of the dominant frequency, p. 52
$i_{L_{\alpha\beta}}$	converter filter ac-current, p. 39

$i^{sp}_{L_{\alpha\beta}}$	ac-current reference, p. 39
$\mathbf{U}^o$	initial population, p. 35
$P_u(\mathbf{u})$	input penalty function, p. 31
N	maximum number of generations, p. 22
N_s	number of switching-states, p. 60
n_g	number of generations, p. 59
n_p	number of predictions, p. 59
$\mathrm{J}^\star$	optimal cost function value, p. 34
$\mathbf{u}^\star$	optimal solution, p. 34
$P_y(\mathbf{y}^p)$	output penalty function, p. 31
$\mathbf{y}^p$	output prediction, p. 30
$\mathbf{y}^{sp}$	output reference, p. 30
M	population size, p. 22, 35
$i^p_{L_\alpha}$	predicted ac-current - α component, p. 40
$i^p_{L_\beta}$	predicted ac-current - β component, p. 40
r_1, r_2	weight-factors of the Jaya-MPC algorithm, p. 37
$\mathbf{u}$	system input, p. 30
$\mathbf{y}$	system output, p. 30
$\mathbf{x}$	system states, p. 30
$\mathbf{u}^{inf}$	inferior limit of the system input, p. 31
$\mathbf{u}_{\mathrm{n+1,m}}$	new individual, p. 22
$\mathbf{u}_{\mathrm{n,m}}$	current individual, p. 22
$\mathbf{u}^{sup}$	superior limit of the system input, p. 31
$\hat{F}_{sw}$	Set of switching frequency components, p. 52
H_{sw}	Set of the switching frequencies' weights, p. 52

$\mathbf{u_w}$	worst solution of the population, p. 35
$\mathbf{y}^{inf}$	inferior limit of the system output, p. 31
$\mathbf{y}^{sup}$	superior limit of the system output, p. 31
$m_{\alpha\beta}$	converter modulation index, p. 39
tol	tolerance, p. 33
v_{dc}^{k}	dc-link voltage, p. 39
$v_{g\alpha\beta}$	grid voltage, p. 39

List of Abbreviations

CCS-MPC	Continuous-Control-Set MPC, p. 9
CCS	Code Compose Studio, p. 63
DG	Distributed Generation, p. 26
EA	Evolutionary Algorithms, p. 21
EMPC	Explicit Model Predictive Control, p. 6
FCS-MPC	Finite-Control-Set Model Predictive control, p. vii
GPC	Generalized Predictive Control, p. 1
IDE	Integrated Development Environment, p. 63
LFC	Load Frequency Control, p. 26
LQR	Linear-Quadratic Regulator, p. 27
MPC	Model Predictive Control, p. vii
OSS-MPC	Optimal-Switching-Sequence-MPC, p. 9
OSV-MPC	Optimal-Switching-Vector-MPC, p. 9
PMSM	Permanent-Magnet Synchronous Machine, p. 28
PSS	power system stabilizers, p. 27
SI	Swarm-Intelligence, p. 21
TFS	Total-Frequency-Spread, p. 53
UPQC	Unified Power Quality Conditioner, p. 27
WECS	Wind Energy Conversion Systems, p. 8

Chapter 1

Introduction

In the last two decades, model predictive control (MPC) stood out as the most promising alternative to linear controllers in power electronics converters [1]. The most common MPC strategy in power electronics is the Finite-Control-Set MPC (FCS-MPC) [2–4]. It presents several advantages: simple implementation and no need for a modulator; fast dynamical response and high disturbance-rejection capability; easy handling with multivariable systems, and easy inclusion of nonlinearities and constraints [1, 4–6]. But, MPC techniques often present also a few drawbacks: elevated computational cost and variable switching frequency.

There are power electronics applications that benefit from variable switching frequency [7], but they differ from the FCS-MPC spread profile: these techniques provide a well-determined, i.e., a known and controlled, spread profile to reduce electromagnetic interference. FCS-MPC variable switching frequency is not well-determined and presents a highly spread profile.

Most trend research regarding MPC in power electronics aims to solve the variable switching frequency [8–17]. Indeed, these researches split into two main paths: adding the switching frequency variability in the optimization problem, whether by adding the switching frequency in the cost function or by selecting an optimal switching sequence aiming at achieving a fixed switching frequency — which typically turns it into a more complex approach, compared to the classical FCS-MPC —, or using another class of MPC, called Generalized Predictive Control (GPC) that solves a continuous-control-set problem, instead of a finite one, and intrinsically provides a fixed switching frequency, by using a modulator. However, this second approach typically presents a complex formulation and implementation and not rarely high computational costs — all rely on the complexity of the optimization solver.

The MPC strategies that aim at providing fixed switching frequency without the use of a modulator do not provide a quantitative analysis of the level of variability

of the switching frequency, i.e., they do not present a metric to quantify how much these methods reduce the variability of the switching frequency to consider them as fixed (or constant). To solve the variability of the switching frequency in FCS-MPC, one may either increase its complexity or change to another strategy that also presents high-complexity and high-cost drawbacks: this is the motivation of this work.

Motivation:

- *FCS-MPC has emerged as an alternative to GPC in power electronics applications due to its simple and very intuitive implementation, but it leads to variable switching frequency;*
- *The works that analyze or attempt to control the switching frequency of FCS-MPC do not provide quantitative tools to calculate the switching frequency spread. Often it is not the aim of these studies, but this observation shows a gap in the field concerning the quantitative analysis of the switching frequency components of MPC approaches.*

Main Question:

- *Is it possible to design a GPC strategy proposing an optimization solver that makes its formulation and implementation simple and reduces the computational cost to values close to the ones observed in FCS-MPC?*

1.1 Proposal Description

This thesis aims to propose an alternative solution to the classical FCS-MPC approach. This proposal intends to provide a viable computational cost with simple implementation, besides a fixed switching frequency.

Including the switching frequency variability in the optimization problem of FCS-MPC naturally increases its computational cost and formulation and implementation complexity. Besides, the literature misses a method to evaluate the reduction of the switching frequency spread. Hence, this work focuses on the alternative of building a GPC solution with an optimization solver that can achieve the desired features: simple implementation and low computational cost. This research also provides a method to quantify the switching frequency variability.

Several works study different algorithms for GPC, but the literature review carried out in this thesis shows that a recent meta-heuristic optimization method, called Jaya algorithm, stands out as a promising alternative.

This thesis proposes an MPC strategy, called Jaya-MPC, to power electronics converters with a fixed switching frequency. This new method combines a modified Jaya algorithm with model predictive control to achieve reduced computational cost and simple implementation while presenting high steady-state and dynamical performances.

So, if the formulation and implementation of the proposed controller are simple enough, close to the simplicity of the classical FCS-MPC, and if it presents a feasible computational cost, it is simpler than the alternatives that increase the formulation complexity of FCS-MPC.

1.2 Methodology

Using the Jaya optimization method as the solver of a continuous-control-set MPC demands few modifications in the original algorithm, which leads to the necessity of four analyses to demonstrate the effectiveness of this new method:

- A parametric analysis to address the changes in the original Jaya algorithm that demand a proper parameter setting. This analysis will lead to a simple approach with only one control parameter;
- An analysis of switching frequency profiles of FCS-MPC. A secondary contribution of this work emerges from this study: a new metric, based on the instantaneous estimation of the switching frequency, that evaluates the total spread of its profile;
- A discussion about the computational cost of both Jaya-MPC and FCS-MPC;
- Experimental validation of the proposed Jaya-MPC controlling a power converter in real-time.

The scope of the analyses of this work is limited to the application of Jaya-MPC in an ac-current control of a three-phase converter connected to the grid, i.e., the proposed control technique is indicated for applications that demand ac-current control with the typical MPC features — fast dynamical response and high-disturbance rejection —, besides a fixed switching frequency. But at the end of this document, it will be explicit that the extension of the proposed MPC in other applications in power electronics is straightforward.

1.3 Work Organization

Chapters 2 and 3 review the state-of-the-art on model predictive control in power electronics and the Jaya algorithm. They present their fundamentals, classifications, and main open topics.

Chapter 4 introduces this thesis proposal: a modified Jaya algorithm used as the solver of a model predictive control applied to power converters. It also addresses the modifications in the original Jaya algorithm.

Chapters 5 discuss the parametric analysis findings that lead to a simple design for the proposed controller.

Chapter 6 presents the secondary contribution of this work: a metric to evaluate the switching frequency spread of power converters operating with model predictive control.

Chapter 7 presents the experimental results that demonstrate the effectiveness of the proposed controller concerning control regulation performance, computational cost, and power quality.

Chapter 8 closes this document by presenting the main findings of the work.

The appendices A, B and C present simulation results regarding disturbance rejection, the use of the original weights of the Jaya-MPC, and an additional discussion about the switching frequency spread of FCS-MPC; these results do not comprise the main scope of this thesis but provide valuable paths for the research and future works.

Publication

Part of the findings of this Thesis have been published in the following paper:

[18] - Tricarico, T.; Costa, J.A.; Herrera, D.; Galván-Díez, E.; Carrasco, J.M.; Aredes, M. **Total Frequency Spread: A New Metric to Assess the Switching Frequency Spread of FCS-MPC**. Energies 2022, 15, 5273.

Chapter 2

Literature Review On Model Predictive Control

Model predictive control is a class of control strategies that predict the system's future behavior based on its dynamical model; It asses the predictions in a cost function to select the best control action that fits the control objectives.

Implementing this control strategy demands three key elements: the predictive model, evaluating the predictions, the cost function, assembling the control objectives and constraints, and the optimization algorithm, solving the optimization problem defined by the cost function [1].

MPC advantages include feasibility for multivariable systems, fast dynamical response, and straightforward incorporation of constraints and nonlinear features [1]. However, this strategy demands high computational effort in comparison with classical linear control — for a long time, only processes with very-slow dynamics were feasible for the use of MPC —, but the advances in microprocessors technology have brought MPC to a new era by enabling its application in systems with fast-dynamics, like power-electronics based systems [1, 19, 20].

2.1 MPC in Power Electronics: A Historical Survey

2.1.1 From 1980 to 2000: The First Proposals of MPC

The first report on the use of MPC in power-electronics dates from 1983 [21] when Joachim Holtz proposed a predictive controller to regulate the stator current of an ac-machine driven by a switched voltage source; and, in [22], Ralph Kennel adds precalculation blocks (predictions) to the standard dc-current control loop of a thyristor bridge.

Although Ralph Kennel calls his proposal a predictive control strategy, his approach differs from the actual comprehension of model predictive control in power electronics; most papers regarding MPC that review the subject claim Joachim Holtz's work was the first MPC application in power electronics.

Before these works, only the process industry field [1] dealt with MPC. After that, in 1994 [23], Hoang Le-Huy and his colleagues implemented a predictive ac-current controller — similar to the one of [21] — in a 1-KW permanent-magnet synchronous motor driven by a PWM-based inverter.

2.1.2 From 2000 to 2010: GPC vs. FCS-MPC

Despite the innovations of [21] and [23], the study of predictive control in power electronics remained out of focus — the MPC field remained restricted to the industry process fields — until the first decade of the 2000s when researches about MPC in power electronics become a trending topic mainly by the works of José Rodrìguez and his colleagues [24–34], and Ralph Kennel and Arne Linder [35–38].

In 2000, Ralph Kennel and Arne Linder first introduced the concept of Generalized-Predictive-Control (GPC) to power electronics: they proposed the application of GPC to control both ac-current and speed of a squirrel cage induction machine fed by a voltage source inverter [36]. The study verified that GPC's performance was superior to that of the PI-based controller. However, the computational cost of GPC was twice higher despite having implemented timesaving precalculation approaches and neglected the cross-coupling terms in the stator current model to simplify the model and reduce the computational cost.

During the 2000s, Ralph Kennel and Arne Linder were the pioneers in the study of MPC techniques for motor-drive control. After introducing GPC in power electronics [36], they proposed the use of Explicit Model Predictive Control (EMPC) to reduce the amount of computational cost associated with GPC by moving part of the mathematical effort offline [37, 38]. However, EMPC demands a very complex mathematical background.

In parallel to the work of Ralph Kennel and Arne Linder, José Rodríguez and

his colleagues followed a different path: instead of considering a continuous control action —inherent to GPC and EMPC formulation —, they assessed the model predictions considering a discrete control set since power-electronic converters actuators present discrete nature.

This idea, presented in 2004 [39], led to an MPC algorithm with a lower computational, simple formulation, and fast dynamical response. Today, this algorithm is known as Finite-Control-Set Model Predictive Control (FCS-MPC) and eliminates the need for a modulator since it directly provides the switching pulses to the power-converter. Yet, this implementation has a drawback: the resultant switching frequency presented a variable profile.

From there on, several works [24–34, 40–47] proposed FCS-MPC in many power electronics applications due to its simplicity and fast response — Indeed, the literature states that, regarding reference-tracking response and disturbance rejection capabilities, MPC outperforms linear control techniques, such as PI and PR based controllers [48–51]; both GPC and FCS-MPC grant faster dynamical response and a stiffer performance when compared to classical linear control.

In 2009, José Rodrìguez and his colleagues provided a detailed description of the fundamentals and applications of FCS-MPC [1] in power electronics and reviewed the most relevant works from the early 1990s until 2009 [34] resuming the state-of-the-art of MPC applied to power electronics converters until then.

2.1.3 From 2010 to 2020: The Growth of FCS-MPC

Since then, works that deal with MPC in power electronics have increased exponentially, as depicted in Figure 2.1 — which compiles the publications in conferences and journals of IEEE and IET regarding MPC in power electronics from 2000 to 2022 — and, in 2011, according to [20], the first MPC-based drive entered the market of electrical-drives. The growth of MPC applications in the last decade is due to the simple implementation, simple formulation, and viable computational cost of FCS-MPC that popularized this technique. These valuable features are always in line with technical advances.

In 2012, José Rodríguez and Patricio Cortés published the first and best-known book on MPC for power converters and electrical drives [51]. Since most works of both authors during the 2000s focused on FCS-MPC, the book deals only with this class of controller; it provides detailed implementation and simulation of several applications of FCS-MPC, covering the practical issues related to it.

In 2013, José Rodríguez and his colleagues published in [19] the state-of-the-art of FCS-MPC in power electronics by summarizing the topics presented in [51].

[1] The first time the term Finite-Control-Set Model Predictive Control was conceived was in [34].

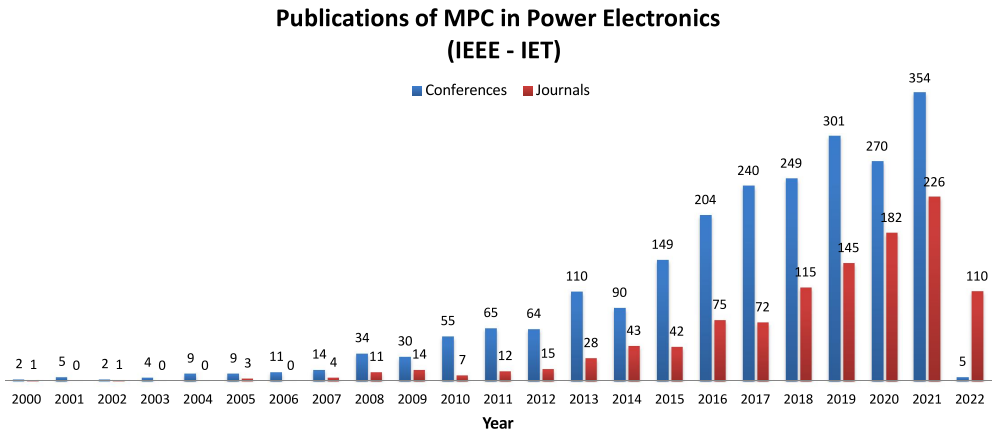


Figure 2.1: Survey of publications regarding MPC applications in power electronics from 2000 to 2022, in journals and conferences of IEEE and IET (survey compiled in June 2022).

In 2015 [20], Samir Kouro, José Rodríguez, and their colleagues discuss the Role of MPC in the evolution of power electronics, compare the features of the main control techniques used in power converters against FCS-MPC, and summarize the most relevant publication of MPC applied to different typologies of power converters in the fields of power quality, machine drive, grid-connected converters, and controllable power supply, from 2008 to 2015.

In 2016, Venkata Yaramasu and Bin Wu Published a book [52] in which they profoundly discuss the applications of FCS-MPC in Wind Energy Conversion Systems (WECS), presenting their fundamentals and the specific features of each predictive control algorithm. Besides, it shows how FCS-MPC became the major trend in MPC applications in power electronics due to its simplicity.

In the same year, Tobias Geyer took a step forward out of academia and compiled the industrial applications of MPC regarding medium-voltage variable-speed drives [53], showing that MPC has already achieved an industrial stage. However, the book also discusses the issues related to the computational burden and the variable switching frequency.

In 2017, Sergio Vazquez, José Rodríguez, and their colleagues published in [1] the most embracing review to date on MPC in power electronics, including GPC and EMPC besides FCS-MPC, which was the main focus of the past surveys presented in [19, 20, 34]. They have summarized the main contributions since 2000 and classified the different MPC strategies in power electronics and their development stages, control diagrams, features, challenges, and open topics.

More recently, in 2021, [54] provides a review of the different types of Model-Free Predictive Control (MFPC) in power electronics applications. This kind of control technique aims at achieving the optimal switching state, often substituting

the predictive model with a look-up table strategy or data-driven model. These approaches present a slightly superior performance to the conventional FCS-MPC under parameter uncertainty. However, the experimental results of [54] only show this improvement when high parameter error occurs — in the order of magnitude of 100% to 150% error of an L filter based converter.

2.2 Classification of MPC in Power Electronics

Power electronics literature divides MPC into two main classes, regarding the type of optimization problem: Finite-Control-Set MPC (FCS-MPC) and Continuous-Control-Set MPC (CCS-MPC) [1, 55].

FCS-MPC takes advantage of the fact that power converters present a discrete nature, i.e., the actuation in these converters is a set of switching pulses called switching states. Hence, the control actions set is the finite set of switching pulses, which leads to a formulation of the integer optimization problem [1, 34]. This approach eliminates the use of a modulator since the control algorithm provides an optimal switching-state (see Figure 2.2 (a)) [1, 34].

On the other hand, CCS-MPC solves the optimization problem to find a continuous control action that minimizes a cost function. Then, a modulator receives the optimal control action to generate the switching pulses (see Figure 2.2 (b)) [1, 55].

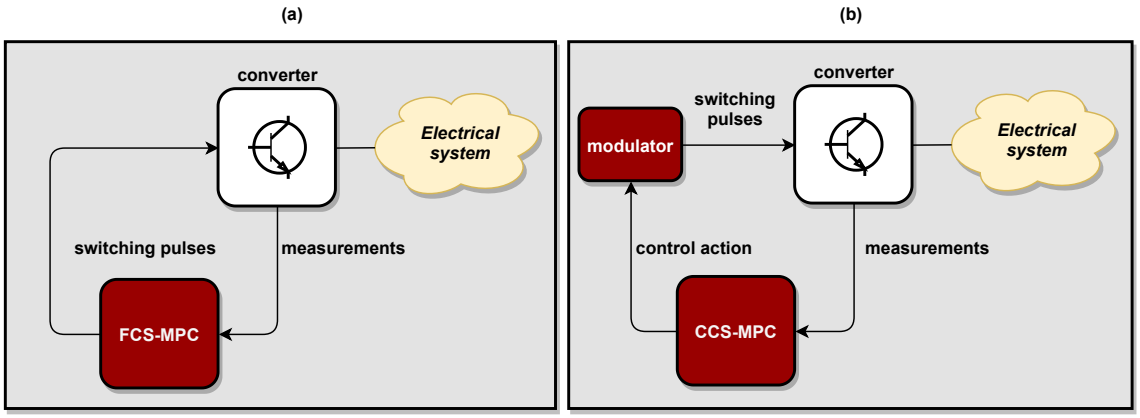


Figure 2.2: General control structure of (a) FCS-MPC and (b) CCS-MPC.

FCS-MPC and CCS-MPC can be divided into two other sub-classifications each (see Figure 2.3): Optimal-Switching-Vector-MPC (OSV-MPC) and Optimal-Switching-Sequence-MPC (OSS-MPC); and Generalized Predictive Control (GPC) and Explicit MPC (EMPC) [1].

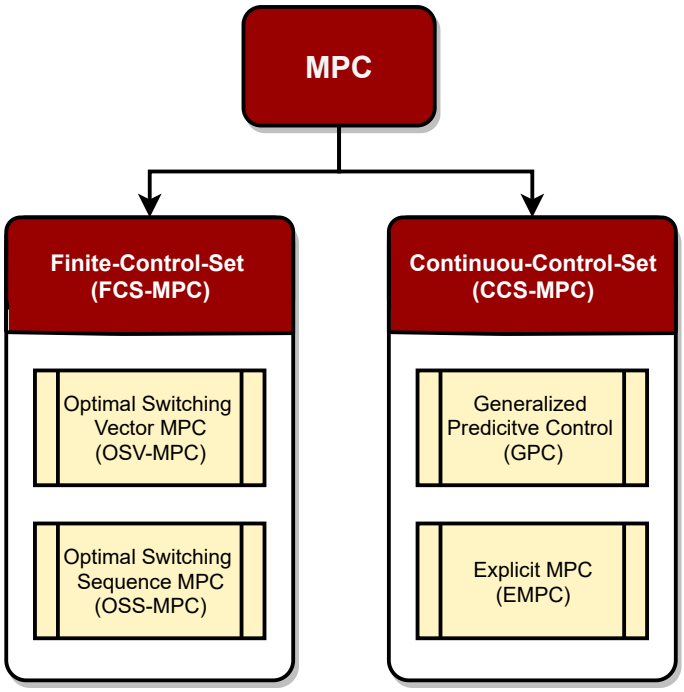


Figure 2.3: Classification of types of MPC applied to power electronics.

OSV-MPC is the first and most common FCS-MPC strategy[2] applied to power electronics converter since it presents the simple formulation and easy addition of constraints [1]. However, the switching frequency of this technique is variable.

OSS-MPC aims at solving the switching frequency variability by considering a set of switching sequences, but it introduces a new variable in the optimization problem: the instant times the switching pulses change state in a switching sequence. This approach gives OSS-MPC the characteristic of emulating a modulator behavior inside the optimization problem; this implementation is more complex than the one of OSV-MPC — it adds several calculation steps that involve derivatives of the cost function regarding the instant times the pulses change [56, 57]. Indeed, OSS-MPC solves the variable switching frequency issues associated with OSV-MPC but increases the computational cost [1], and removes the most attractive feature of OSV-MPC: simplicity — due to its simplicity, OSV-MPC is much more popular MPC approach in power electronics.

GPC is the most common strategy in CCS-MPC because it implements online optimization in contrast with the EMPC, which uses offline optimization and parametric search. Both provide the control action to a modulator, which provides a fixed switching frequency profile. However, both strategies present complex formulations [1].

EMPC solves offline optimization problems for several configurations and stores

[2]Since OSV-MPC is the first FCS-MPC strategy used in power electronics, it is also referred to as FCS-MPC in the literature.

the solutions in a table format, resulting in an embedded online control that only searches this table using search algorithms. Although this strategy reduces the computational cost, it needs high amounts of memory, which elevates the total cost of the solution [1].

Analytically solving part of the optimization problem may reduce the computational cost of GPC, but it is only simple for reduced-order problems with linear models [1].

Both GPC and EMPC can add constraints and non-linear features to the optimization problem, but, in the case of GPC, it increases the computational cost [1, 55]. These techniques may implement long horizon predictions, but it raises the computational cost, demands enhanced search algorithms for FCS-MPC strategies, and grows the order of the optimization problem of CCS-MPC. Hence, it is more common to use a unit prediction horizon.

Both OSV-MPC and GPC are the most popular approaches among these techniques, but, in power electronics applications, OSV-MPC is the most used technique since 2004 [3].

[3]The interest of the field in GPC has reduced since the FCS-MPC proposal because of the simplicity and intrinsic feasibility that FCS-MPC provides to power converters.

2.3 Issues Related to MPC

Each MPC algorithm is different, depending on the application, the control objectives, and the type of MPC strategy, but they share three common elements: the predictive model, the cost function, and the optimization solver. Each of these elements presents different issues, and most of the studies in the power electronics literature aim to address these issues [1, 51].

2.3.1 Regarding The Predictive Model

The predictive model is the element that describes the system dynamics over the prediction horizon. Because of the digital nature of MPC implementation, the predictive model is discrete and obtained by using the Euler approximation and Taylor series if the system is linear. However, the discretization of nonlinear systems often demands other techniques, which may be more complex [1].

The quality of the prediction model, therefore, the quality of the discretization plays a significant role in the predictive controllers' performance, i.e., the control effort will only be an optimal solution if the predictions are precise. The main applications of MPC use the *Forward-Euller* method to obtain the predictive model.

In FCS-MPC, the imprecision of the predictive model may degrade its performance even more since it presents variable switching frequency, which may push the converter to resonance problems [1].

Several works propose solutions for mitigating resonance problems using an active damping filter or a long-prediction horizon strategy; both increase the computational effort [58–60].

2.3.2 Regarding The Cost Function

Building the cost function — and setting its weight factors, if that is the case — is the main task in MPC: it defines the control objectives that the controller aims at achieving and influences steady-state characteristics, like RMS value of ac electrical quantities, total-harmonics-distortion (THD), and harmonic spectrum [1]. For instance, in [51], the authors study this influence by comparing two simple cost functions with the same current regulation objective and demonstrate that quadratic-error ($||e||^2$) and the absolute value of the error ($|e|$) present different steady-state waveforms in a three-phase converter.

Figure 2.4 presents the main control objectives and the most common secondary goals found in the literature regarding MPC in power electronics.

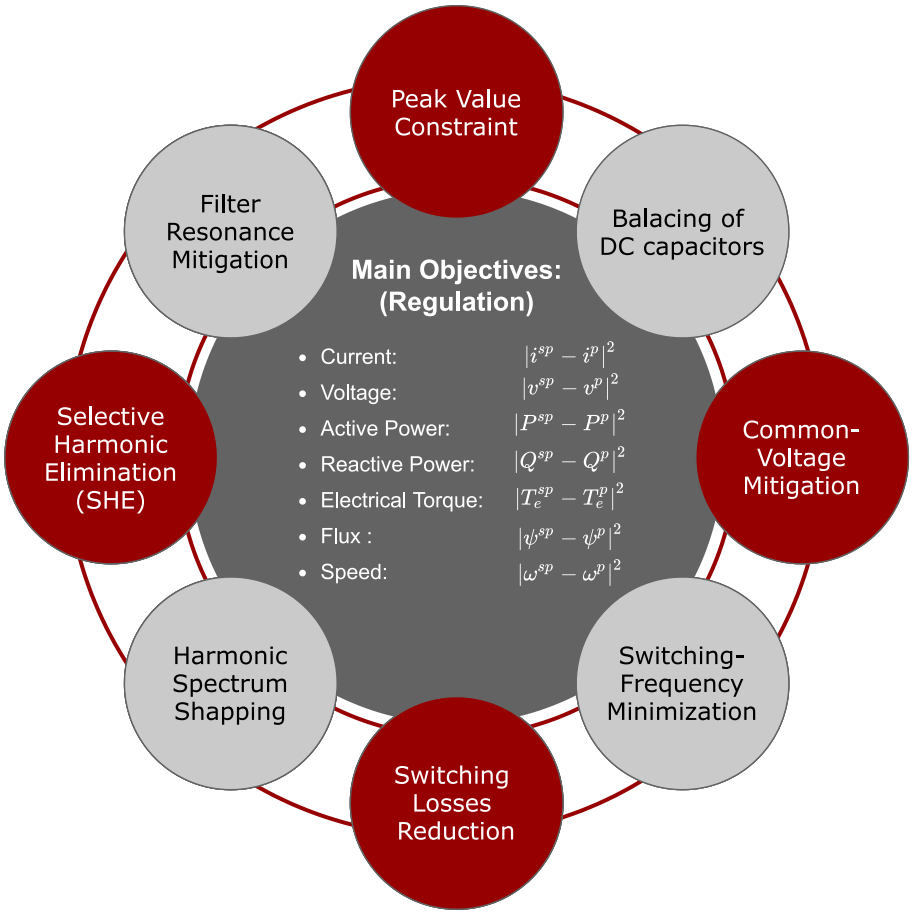


Figure 2.4: Main and secondary control objectives allowed by MPC in power electronics applications.

The main control objectives often comprise a quadratic-error cost function of one or more variables, e.g., currents, and voltages [1, 19, 34, 51]. However, it is often common to find other objectives related to torque, flux, speed, and active and reactive power in applications such as control of electrical machines [52].

On the other hand, secondary objectives are suitable for penalties or added as additional costs, weighted by proper weight factors [52].

Adding peak-value constraints in both CCS-MPC and FCS-MPC is one of the most common secondary objectives. Some works related to OSV-MPC include variables in the cost function to minimize the number of commutations in each sampling period to reduce losses or the variability of the switching frequency [1, 30], but these additions make the cost function and its weighting design more complex.

In GPC and EMPC, these variables are unnecessary since they use a modulator that produces fixed switching frequency [1]; it is also possible to add the variation in the control effort in the cost function to reduce its slew rate, although it is not common in power electronics applications [1, 19].

When the cost function deals with a multi-objective problem, it often uses

weighting-factors [1, 19]. The design of these factors affects performance, robustness, and stability of the controller — the literature already established that optimal control does not guarantee system stability [1]—, but only a few works deal with stability demonstration of MPC in power electronics systems [61–63].

It is well-established that using a per-unit base system helps the decision of the weight factors values [33]. However, optimal weighting factor selection is still an open topic in MPC [1].

2.3.3 Regarding The Optimization Solver

The main challenge of the optimization solver is finding an optimum solution within the time of a sampling period. In power electronic applications, this period is around a few microseconds. Thus, not every algorithm can be used in MPC, making this issue the main aim of several works [1] — including this thesis.

The computational burden of a predictive controller is strongly related to the number of times the algorithm evaluates the cost function and the predictive model. Hence, optimization algorithms that achieve the optimal solution with a reduced number of evaluations tend to present a lower computational cost [1].

Some approaches of GPC try to calculate an approximation of the optimum solution to reduce the number of generations of the optimization algorithm [55, 64]. However, it may lead to local-optimum, depending on the cost function, and, in a non-linear system, it may not even be possible [1].

In FCS-MPC, some integer-optimization techniques reduce the computational effort by disregarding the redundant switching states, thus reducing the number of predictive models- and cost function- evaluations [65–68]. Although several studies focus on reducing the computational burden, it is still an open topic in MPC applied to power electronics converters [1].

2.3.4 Regarding The Switching Frequency

FCS-MPC and its variations have been the subject of several types of research in recent years, applied in various branches of power electronics [1], [2], [5] and [6], due to the flexibility that the cost function provides in controlling the variables of interest, as well as presenting a superior transient response to linear control [69].

According to [8], the variable switching frequency is one of the open problems of FCS-MPC. This issue is associated with a non-symmetric and widely spread frequency spectrum that can cause resonance problems, making FCS-MPC unsuitable for applications that require strict harmonic control, such as in grid-connected converters or powering sensitive loads [4].

Several proposals aim to mitigate or control the FCS-MPC switching frequency, but so far, the state-of-the-art of MPC in power electronics misses quantitative analyses regarding the switching frequency spread profile of FCS-MPC.

Spread-spectrum modulation techniques (SS-PWM) are a solution for reducing electromagnetic interference (EMI) reduction [7]. These modulation methods shape the switching frequency spectrum in such a way that the energy of the modulated signal distributes spreads in a predetermined range rather than concentrating it on a single frequency.

These spread-spectrum techniques differ from FCS-MPC since they use a modulator with a known modulation strategy that distributes the switching frequency within a predetermined range and shape, whereas FCS-MPC produces a variable, widely spread, asymmetric, and not well-defined switching frequency profile. Table 2.1 provides a brief comparison of the variable switching frequency of SS-PWM techniques and FCS-MPC.

Table 2.1: Comparison between the variable switching frequency characteristics of FCS-MPC and the SS-PWM techniques.

	SS-PWM	FCS-MPC
symmetry	symmetrical spread	asymmetrical spread
controllable	controlled bandwidth	uncontrolled
profile shape	well-determined shape	not fixed shape, highly dependent of the parameters of the controller
drawback	more complex implementation than conventional PWM	may cause resonance problems, zero crossing issue, and low power quality
advantage	EMI reduction	-

Metrics to Assess Switching Frequency

Up to now, the only metric to characterize the switching frequency of FCS-MPC is the average switching frequency [3].

Equation (2.1) gives the value of the average switching frequency, $\bar{f}_{sw}$, over a period equals MT_s, but this quantity does not estimate the variability,i.e., the spread of the switching frequency.

$$\bar{f}_{sw} = \lim_{M\to\infty} \frac{1}{n_s c_k M T_s} \sum_{m=1}^{M} \|\Delta S_m\|_1 \tag{2.1}$$

M is the total of samples, T_s is the sampling period; ΔS_m is the difference between two consecutive switching states; n_s is the number of power switches per leg in the power converter under analysis; and c_k is a correction factor that depends on the converter topology and aims to accomplish the criteria of (2.2). For instance, in a two-level three-phase converter, both c_k and n_s are equal to 2.

$$\frac{\Delta S_m}{c_k} \in \{-1, 0, 1\} \tag{2.2}$$

Another metric found in the literature to characterize the power quality of FCS-MPC is the Total Harmonic Distortion (THD). However, the THD calculation only considers the harmonic components of the fundamental frequency, i.e., it does not quantify the non-characteristic components of the switching frequency.

One of the first efforts to control the switching frequency in FCS-MPC was described in [8] and [9]. These works added a term in the cost function to penalize excessive switching state changes. Yet, this method guarantees neither a fixed switching frequency nor a well-defined frequency spectrum.

Further, [10] and [11] attempts to shape the switching frequency spectrum by mitigating specific harmonics.

The work of [10] uses a notch filter in the cost function to avoid specific harmonic components. Even though this approach can reduce some harmonics, the switching frequency is still variable.

The study of [11] uses a similar method based on a sliding discrete Fourier transform (SDFT) method applied to the cost function. This approach calculates a finite number of components of distinct frequency spectra, assigning weights for each harmonic. However, as in [10], the suppression of some harmonics does not ensure a fixed switching frequency. Both works increase the formulation complexity of FCS-MPC and come up against the fact that they do not know the non-characteristic frequencies produced by FCS-MPC.

The work of [70] proposes a hybrid OSV-MPC by adding a low-pass filter to remove the high-frequency components in the switching function produced by a classical FCS-MPC. But this approach removes one of the key features of FCS-MPC: the fast dynamical response because the actuation signal — the filter output — presents slow dynamics due to the low cut-off frequency. Also, the results show a high ripple and do not discuss the effect of adding the filter in the regulation error.

The works of [13] and [14] use the concept of OSS-MPC — which formulates the optimization problem as a function of time instants the switching vectors transit. In both cases, the MPC technique involves calculating the derivative of the cost function besides demanding a switching sequence generator block, which increases the computational cost and the formulation complexity. These are valuable advances

in the state-of-the-art and contribute to providing a solution that mimics the behavior of a modulator strategy; the results of the method of [13] do not discuss the switching frequency profile produced by OSS-MPC.

On the other hand, the methods defined in [15],[71] and [16] adds the modulation stage in the cost function. This method — called modulated MPC (M^2PC) — is similar to the OSS-MPC since it also needs an additional stage to calculate the switching times. [15] and [71] use this technique for a multilevel-converter while [16] uses it for an NPC converter. These works are also valuable advances to the state-of-the-art, but despite mimicking de behavior of a modulator stage and reducing the THD — compared to FCS-MPC —, they yet produce a spread profile when analyzing the frequency spectrum using FFT. These approaches add extra calculation steps that increase the formulation complexity compared to FCS-MPC, like those of the OSS-MPC class.

Two recent papers, [72] and [73], study how different cost function strategies affect a grid-forming converter, but both miss a more rigorous quantification of switching frequency spread.

The work of [72] examines the influence of two cost functions without frequency control and two cost functions with frequency control in the voltage THD at several operating points. However, the analysis of the cost functions' impact on the switching frequency is only qualitative.

The study presented in [73] in 2020 expanded the findings of [72], performing an in-depth examination of the cost function's weights and sampling frequency impact on switching frequency and THD.

However, [73] bases its analysis only on average switching frequency , like other works in the state-of-the-art [8], [9], [11]. Other works [10], [12], [13], [14], [16] do not use this metric, assessing only THD and FFT.

All of those works use different techniques, aiming to control the FCS-MPC switching frequency profile and provide valuable contributions to the field. However, the state-of-the-art does not fully assess the spread profile of the switching frequency of FCS-MPC approaches. Most of them use only the average switching frequency to validate the effectiveness of their analyses. However, this thesis will demonstrate, in Chapter 6, that this average metric cannot fully characterize the switching frequency of FCS-MPC.

2.4 The Current Dilemma of MPC in Power Electronics

The literature review on the state-of-the-art of MPC in power electronics leads to some statements regarding the challenges that MPC faces:

- FCS-MPC has developed as a viable and simple-to-implement alternative to GPC over the past two decades because of the infeasibility of GPC concerning computational cost and the adaptability of FCS-MPC to various power converters topologies;
- Although FCS-MPC presents variable switching frequency, it is the most used predictive controller because of its simplicity;
- In FCS-MPC, the cost function can include switching frequency as a control objective, but it results in a more complex control problem;
- Besides, including multi-objectives in the cost function hampers the weight-factors selection;
- Other strategies derived from FCS-MPC, like OSS-MPC, try to control the switching frequency by assessing the switching time instants in the optimization problem or even emulating the behavior of a modulator;
- Although these techniques reach relative success regarding the qualitative reduction of switching frequency spread, they increase the formulation complexity and computational cost in relation to FCS-MPC;
- All these works do not assess the switching frequency spread quantitatively, only measuring the average switching frequency;
- GPC is a solution with fixed switching frequency and online solving of the optimization problem — instead of EMPC that solves it offline;
- However, the implementation complexity of GPC is higher than FCS-MPC since solving a continuous optimization problem is often more difficult than solving an integer optimization problem and demands more computational effort;
- In summary, FCS-MPC gained attention because of its simplicity; OSS-MPC techniques attempt to fix the switching frequency variability of OSV-MPC at the expense of more complex and computationally costly approaches, while GPC has been left aside in the mainstream of power conversion applications.

These observations point to the question that motivated this work:

Is it possible to design a GPC strategy proposing an optimization solver that makes its formulation and implementation simple and reduces the computational cost to values close to the ones observed in FCS-MPC?

Chapter 3

Literature Review On Jaya

This chapter reviews a recent meta-heuristic algorithm called Jaya and its applications in power electronics. It describes how this algorithm works, showing its main advantage: simple implementation. At the end of the review, it became clear the gap in the application of this algorithm in MPC; this thesis intends to explore this gap.

3.1 Background

Classical optimization techniques are well-established in several applications in engineering, providing mathematically well-grounded solutions. However, these methods may fail when applied to more complex optimization problems. In these cases, meta-heuristic techniques stand out as an alternative that is commonly more flexible and efficient, besides dismissing the necessity of first and second-order derivatives. Yet, these techniques present a higher computational cost and several algorithm-specific control parameters, called hyper-parameters.

With the advances in computational hardware capacity over the decades, the shortcomings related to the computational cost tend to decrease, leading to increased use of meta-heuristic methods — mainly in offline optimization problems. Even so, the difficulty of tuning hyper-parameters continues to be a drawback since these parameters strongly affect the performance of their respective optimization methods.

Natural processes like evolution commonly inspire meta-heuristic optimization techniques — the literature also refers to them as natural optimization.

Two groups of algorithms classify most of the meta-heuristic methods: Evolutionary Algorithms (EA) and Swarm-Intelligence (SI) based algorithms. While the former tries to mimic natural evolution (natural selection) rules to solve optimization problems, the latter metaphors biological group-behavior observed in nature.

These methods are probabilistic (non-deterministic) and require usual hyper-parameters like population size and the number of generations. Besides these, each algorithm requires its specific hyper-parameters. The precise tuning of these algorithm-specific hyper-parameters is a critical factor that affects the performance of each algorithm. The improper tuning may increase the computational effort or even jeopardize the search for an optimal solution. Online control applications avoid these algorithms because of the complexity, probabilistic nature, and difficulty of setting hyper-parameters.

3.2 Jaya Algorithm

In 2015, Venkata Rao proposed in [74] a new meta-heuristic optimization method, called Jaya Algorithm, which presented the advantage of simple implementation and dismissed algorithm-specific hyper-parameters. These assets and the algorithm's performance drew the attention of researchers in the field of meta-heuristic optimizations. It is also feasible for multi-objective and multi-dimensional optimization problems [75].

In the original work [74], the author compares the performance of the Jaya algorithm against a well-established group of optimization algorithms by testing them

in a benchmark composed of 24 constrained and 30 unconstrained optimization problems. The statistical results of the tests have also supported the superior performance of the Jaya method. Yet, the authors emphasize that neither Jaya nor any other algorithm can be referred to as the 'best' algorithm.

Jaya is a population-based algorithm similar to well-established meta-heuristic algorithms, but it differs from them regarding the inspiration, as it does not mimic any natural process. Indeed, the core of the Jaya algorithm involves a simple rule: getting closer to the optimal solution and moving away from the worst solution, i.e., it calculates the new individuals ($\mathbf{u}_{\mathrm{n+1,m}}$) of the population based on the best ($\mathbf{u_b}$) and the worst ($\mathbf{u_w}$) individuals of each generation.

Equation (3.1) implements this core idea, in which the indexes m and n means the m^{th} individual of the population at the n^{th} generation; and the parameters r_1 and r_2 are random-variables, which weight both the terms $(\mathbf{u_b} - |\mathbf{u}_{\mathrm{n,m}}|)$ and $(\mathbf{u_w} - |\mathbf{u}_{\mathrm{n,m}}|)$ that aims to get the new individual closer to the best solution and to move it away from the worst solution.

$$\mathbf{u}_{\mathrm{n+1,m}} = \mathbf{u}_{\mathrm{n,m}} + \mathrm{r}_1 \left(\mathbf{u_b} - |\mathbf{u}_{\mathrm{n,m}}|\right) - \mathrm{r}_2 \left(\mathbf{u_w} - |\mathbf{u}_{\mathrm{n,m}}|\right) \tag{3.1}$$

If the new individual of a population is a better solution than the previous one, the algorithm includes it in the next one; if not, the algorithm discards the new individual, keeping the previous one in the next generation.

Algorithm 1 presents the procedures of the Jaya optimization method, and Figure 3.1 depicts its flowchart. Note that it only demands the setting of two common hyper-parameter: population size (M) and maximum number of generations (N). It sets a random initial population and does not require any algorithm-specific hyper-parameter since it generates the random weights (r_1 and r_2) with uniform distribution.

After the publication of the original Jaya algorithm, many works developed variant methods applied to engineering optimization problems [76–80]. These variants modify the original procedure according to the problem they aim to solve.

In 2019, Venkata Rao published the only book [81] so far, compiling and describing the applications and procedures of the Jaya algorithm and its variants up to the end of 2018. The book reports a few applications in electrical engineering, but none related to model predictive control.

Algorithm 1: Original Jaya algorithm (introduced in [74]).

Data: $f : U \to V$; $\mathbf{u} \in \mathbb{R}^n$; $\mathrm{v} \in \mathbb{R}$;

Result: optimal solution $\mathbf{u}^\star = \underset{\mathbf{u}}{\operatorname{argmin}}\, \mathrm{f}(\mathbf{u})$; optimal cost $\mathrm{v}^\star = min\left(\mathrm{f}\left(\mathbf{u}\right)\right)$

initialize:
    population size (M)
    number of generations (N)
    random initial population set ($\mathbf{U}^o = \{\mathbf{u}_{0,1}, \ldots, \mathbf{u}_{0,\mathrm{M}}\}$)
begin
    select best and worst solution: $(\mathbf{u_b}, \mathbf{u_w})$
    foreach *generation:* $n \leq \mathrm{N}$ **do**
        foreach $\mathbf{u}_{\mathrm{n,m}} \in \mathbf{U}^n$ **do**
            generate random numbers:($\mathrm{r}_1 \sim U(0,1)$ and $\mathrm{r}_2 \sim U(0,1)$)
             evaluate new individual: $\mathbf{u}_{\mathrm{n+1,m}}$
             $\mathbf{u}_{\mathrm{n+1,m}} = \mathbf{u}_{\mathrm{n,m}} + \mathrm{r}_1 \left(\mathbf{u_b} - |\mathbf{u}_{\mathrm{n,m}}|\right) - \mathrm{r}_2 \left(\mathbf{u_w} - |\mathbf{u}_{\mathrm{n,m}}|\right)$
            if $\mathbf{u}_{\mathrm{n+1,m}}$ *is* ***not*** *a better solution than* $\mathbf{u}_{\mathrm{n,m}}$ **then**
                $\mathbf{u}_{\mathrm{n+1,m}} = \mathbf{u}_{\mathrm{n,m}}$
            end
        end
        select best and worst solution: $(\mathbf{u_b}, \mathbf{u_w})$
    end
    return result $\mathbf{u}^\star = \mathbf{u_b}$
end

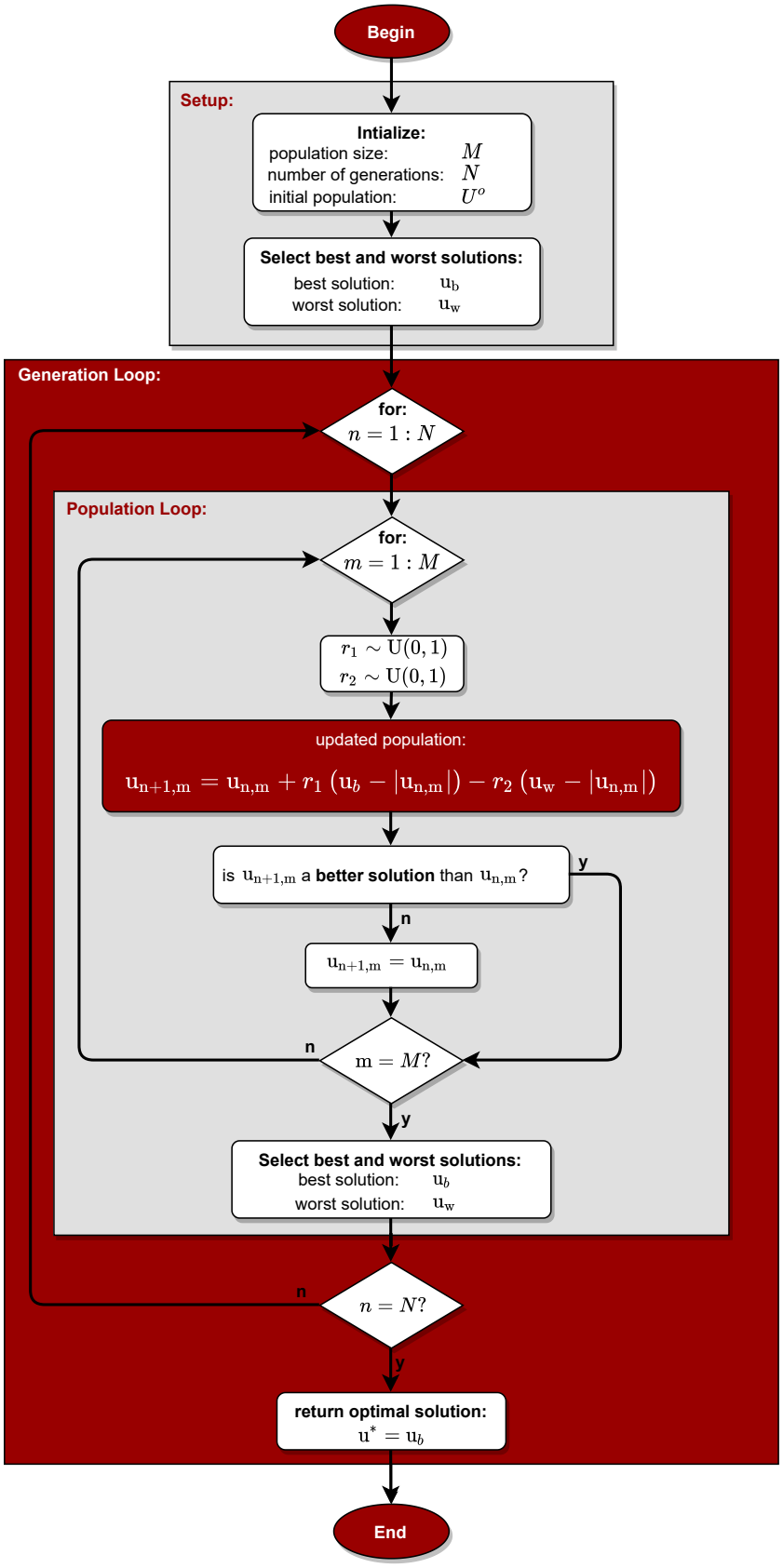


Figure 3.1: Flowchart of the original Jaya algorithm.

3.3 Applications in Power Electronics

Figure 3.2 shows a survey of the publication of the Jaya algorithm in conferences and journals from IEEE and IET since 2016, covering power electronics applications. These data confirm that the Jaya algorithm is the topic of only a few works in power electronics, showing its novelty.

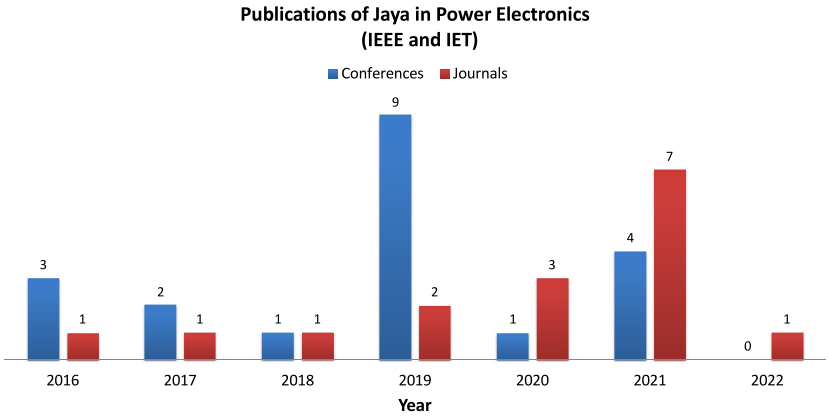


Figure 3.2: Survey of publications covering Jaya algorithm applications in Power Electronics from 2016 to 2022, in journals and conferences of IEEE (survey compiled in June 2022).

Table 3.1 presents a list of the most significant publications of the Jaya algorithm and its variants applied to power electronics. Most applications comprise offline optimization problems, such as control-parameters optimization and optimal power dispatch. A few promissory works use Jaya as an online solver in MPPT algorithms applied to photovoltaic systems. Indeed, none of these works proposes the Jaya algorithm as an offline or online optimization solver for MPC applications; this is the gap this thesis proposal intends to explore.

Table 3.1: Main publications using the Jaya algorithm in power electronics.

Year	Optimal Power Dispatch	Control Parameter Optimization	MPPT algorithms
2016	–	[82]	–
2017	[83]	[84, 85]	[86, 87]
2018	[88–91]	[92, 93]	[94]
2019	[95–99]	[100]	[101]
2020	[102]	–	–
2021	[103, 104]	–	[105, 106]

3.3.1 Applications in Power Dispatch and Energy Management

Many papers present studies regarding the usage of Jaya in optimal reactive power dispatch algorithm [83, 88, 89, 95] — indeed, it is the most common application so far.

In [96], the work assesses an economic dispatch problem in a microgrid with the presence of solar and wind generations. And, in [97], it presents a method based on the Jaya algorithm to find the optimal placement of a D-STATCOM in distribution networks to reduce the total power losses and improve the voltage profile.

The work presented in [90] introduces a new meta-heuristic algorithm, called Quasi-Oppositional-Jaya (QO-Jaya), for solving optimal power flow problems. This proposition is more complicated than the classical Jaya algorithm, but the simulation results lead to the conclusion that the proposed method betters the original Jaya algorithm in reducing the active power losses and improving voltage stability. However, it considers only a few simulation scenarios.

In [98], it studies a probabilistic approach for microgrids' optimal energy management considering ac network constraints, using the Jaya algorithm as the optimization solver. Related research presented in [91] applies Jaya to energy management systems in smart grids. The study published in [99] aims to minimize the losses in a radial distribution system. In this application, Jaya assesses both placement and sizing of Distributed Generation (DG) units.

3.3.2 Applications in Control Parameters Optimizations

The work presented in [100] applies Jaya to enhance the automatic generation control of two power system areas. Simulation results demonstrate that the Jaya approach reduces the frequency deviations of both areas when compared to the case that uses the Ziegler-Nichols method to tune the controller's parameters.

In [84], it uses Jaya to tune the parameters of a speed controller of a DC-motor to improve its dynamical response; it also compares its performance against the one of a PSO (Particle Swarm Optimization) algorithm. The experimental results lead to the conclusion that, while the PSO algorithm presents the best steady-state response, the Jaya algorithm gives the best transient response, although the work studies only one scenario.

In [82], it proposes a new Load Frequency Control (LFC) strategy, based on a fuzzy controller, to decrease frequency oscillations caused by load disturbances and wind fluctuations in a wind-thermal power network. The work applies an online Jaya algorithm to tune the control parameters optimally. A similar approach [93] uses Jaya to design an LFC to mitigate frequency deviations in a multi-area power

system's interconnection problem that incorporates renewable energy generations.

The work presented in [85] uses the Jaya algorithm to tune a Linear-Quadratic Regulator (LQR) employed on a power system stabilizers (PSS) to stabilize a synchronous machine. But, the study provides only a few simulation results comparing LQR with a PID controller.

In [92], the study introduces a Jaya-based method for both shunts- and series-controls of a Unified Power Quality Conditioner (UPQC) applied to a PV system. The optimization problem is based on auto-tuning PR controllers and comprises two cost functions to improve power quality for both voltage and current. It provides experimental results and compares the proposed method with other optimization techniques. The work analyses multiple scenarios, including linear, nonlinear, and unbalanced loads; voltage-sag, voltage-swell, and unbalance-voltage-sag conditions. The results validate the effectiveness of the proposed Jaya method and support the conclusion that the use of Jaya enhances the dynamical response of the system.

3.3.3 Applications in MPPT Algorithms

In [86], it proposes a Jaya-based maximum-power-point-tracking (Jaya-MPPT) method for photovoltaic arrays working under variable shading conditions. In [94], the authors extend this study and provide experimental results to support the conclusion that the proposed Jaya MPPT algorithm beats a PSO-based MPPT algorithm in three key metrics: convergence speed, oscillations in the convergence, and overall tracking efficiency.

The work presented in [87] combines the Jaya algorithm with a Differential Evolution algorithm — called Jaya-DE — to improve the performance of MPPT in photovoltaic systems under high fluctuations in temperature and radiance. The study is very detailed, and the experimental results demonstrate that the Jaya-DE method outperforms other strategies described in the state-of-art of MPPT algorithms.

In [101], the work studies the performance of a Jaya-based MPPT algorithm applied to a hybrid photovoltaic-fuel-cell for grid integration. It provides experimental results — carried out in a hardware-in-the-loop environment — that support the conclusion that Jaya-MPPT presented a performance superior to PSO (Particle Swarm Optimization) or ABC (Artificial Bee Colony) algorithms.

3.3.4 Others Applications

Parameters identification using Jaya is still a topic in power electronics applications: both papers presented in [107] and [108] use Jaya as a parameter identification algorithm for Permanent-Magnet Synchronous Machine (PMSM) and photovoltaic models, respectively.

In [109], it applies a modified-Jaya algorithm in an optimization problem concerning the coordination of overcurrent relays. The results show that the modified-Jaya algorithm outperformed the basic genetic algorithm. However, simulation tests do not include any other optimization.

Chapter 4

Thesis Proposal

Solving a continuous-set optimization problem with low-computational cost in the context of a GPC is the principal task of this work to provide a viable solution for MPC with fixed switching applied to real-time ac-current controlling in a power converter. Thus, this chapter proposes the combination of a modified-Jaya algorithm and GPC. It also details the development of this control strategy and defines the scope of its application. This solution consists of the main contribution of this thesis.

4.1 Jaya Algorithm Applied to MPC

Figure 4.1 shows the control structure proposed in this thesis; the Jaya-MPC block includes the common elements of a typical MPC controller, such as a predictive model, the control objectives, the constraints — defining the control optimization problem —, and the Jaya-MPC algorithm with the modifications (highlighted in red) in the original Jaya algorithm so that it fits as the solver for MPC. In this diagram, the system's input **u** is the power converter's modulation index.

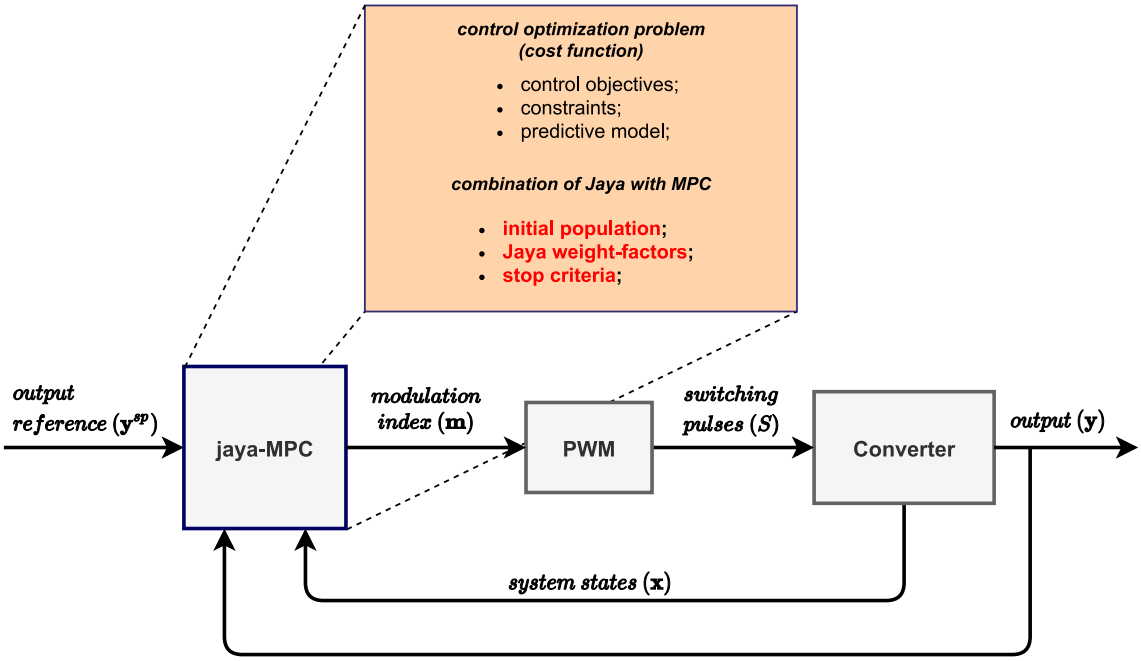


Figure 4.1: Top-level block diagram of the closed-control loop with the proposed Jaya-MPC applied to power converters.

This proposal falls into the GPC class and uses Jaya as the optimization solver. This combination merges characteristics of both GPC and Jaya: fixed-switching frequency (GPC), reduced computational cost, simple implementation, and reduced control parameters in the optimization solver (Jaya).

4.2 Cost Function

The cost function defines the control objectives in model predictive controllers. It may be a purely regulation problem or a constrained[4]-regulation problem.

[4]Simple inclusion of constraints in the control law is one of the most attractive features of MPC [1].

Equation (4.1) shows the general form of the cost function that suits the application of this work. It comprises the quadratic error [5] between the predictive output ($\mathbf{y}^p$) and its reference ($\mathbf{y}^{sp}$) and two penalty terms associated with both inputs and output constraints, respectively. In power converters, the inputs ($\mathbf{u}$) are often the modulation indexes, and the outputs ($\mathbf{y}$) are the electrical quantities the controller aims to regulate, e.g., voltages, currents, active and reactive power.

$$J\left(\mathbf{u}, \mathbf{y}^p\right) = ||\mathbf{y}^p\left(\mathbf{u}\right) - \mathbf{y}^{sp}||_2^2 + P_u\left(\mathbf{u}\right) + P_y\left(\mathbf{y}^p\right) \tag{4.1}$$

4.2.1 Regulation Cost

The first term of the left-hand side of Equation (4.2) deals with regulating the system's output, i.e., the control variable of the power converter; if only this term is active, the optimization problem is free of constraints, and the minimum value of cost function is zero. However, in that case, the Jaya-MPC stops and returns a solution when this cost is less than a pre-defined tolerance.

$$||\mathbf{y}^p\left(\mathbf{u}\right) - \mathbf{y}^{sp}||_2^2 \leq tol \tag{4.2}$$

4.2.2 Input Constraints Penalty

In power converters control, the modulation index bound-limits are the most common input-constraint; Equation (4.3) formally defines these constraints.

$$\mathbf{u}^{inf} \leq \mathbf{u} \leq \mathbf{u}^{sup} \tag{4.3}$$

Equation (4.4) formalizes a typical penalty Function ($P_u\left(\mathbf{u}\right)$) strategy to deals with the inputs constraints:

$$P_u\left(\mathbf{u}\right) = \begin{cases} 0 & , \mathbf{u}^{inf} \leq \mathbf{u} \leq \mathbf{u}^{sup} \\ P & , \text{otherwise} \end{cases} \tag{4.4}$$

4.2.3 Output Constraints

The output constraints play a role in avoiding the control variables, such as voltages and currents, being out of a safe operating range during transient dynamics. In other words, the output penalty term in the cost function acts to make the system operate within the range represented by Equation (4.5).

[5] Studies report that quadratic error cost as a better choice of regulation cost function than the absolute value of the error [51].

$$\mathbf{y}^{inf} \leq \mathbf{y}^{p} \leq \mathbf{y}^{sup} \tag{4.5}$$

Similar to the case of the input constraints, Equation (4.6) defines the output constraints penalty function. This function penalizes the control actions that result in predictions out of the bounds of Equation (4.5).

$$P_y\left(\mathbf{y}^{p}\right) = \begin{cases} 0 & , \mathbf{y}^{inf} \leq \mathbf{y}^{p} \leq \mathbf{y}^{sup} \\ P & , \text{otherwise} \end{cases} \tag{4.6}$$

4.3 Control Optimization Problem

Equation (4.7) formalizes the optimization problem related to the control problem that Jaya-MPC aims at solving; it is composed by the cost function — which depends on the control input ($\mathbf{u}$), the output predictions ($\mathbf{y}^{p}$) and the output reference ($\mathbf{y}^{sp}$) — subject to the model dynamics (predictive model) and the input and the output constraints $\left(\mathbf{u}^{inf}, \mathbf{u}^{sup}, \mathbf{y}^{inf}, \mathbf{y}^{sup}\right)$.

The predictions ($\mathbf{y}^{p}$) are a function (F) of the system inputs ($\mathbf{u}$) and the system states ($\mathbf{x}$).

$$\begin{aligned} \mathbf{u}^{\star} = \quad & \underset{\mathbf{u}}{\operatorname{argmin}}\, J\left(\mathbf{u}, \mathbf{y}^{p}, \mathbf{y}^{sp}\right) \\ \text{s.t.} \quad & \mathbf{y}^{p} = F\left(\mathbf{x}, \mathbf{u}\right) \\ & \mathbf{u}^{inf} \leq \mathbf{u} \leq \mathbf{u}^{sup} \\ & \mathbf{y}^{inf} \leq \mathbf{y}^{p} \leq \mathbf{y}^{sup} \end{aligned} \tag{4.7}$$

If the predictive model is linear, which is a usual case in power converter models that deal with only the filter dynamics, the optimization problem turns into Equation (4.8).

$$\begin{aligned} \mathbf{u}^{\star} = \quad & \underset{\mathbf{u}}{\operatorname{argmin}}\, J\left(\mathbf{u}, \mathbf{y}^{p}, \mathbf{y}^{sp}\right) \\ \text{s.t.} \quad & \mathbf{x}^{k+1} = A\mathbf{x}^{k} + B\mathbf{u}^{k} \\ & \mathbf{y}^{p} = C\mathbf{x}^{k+1} \\ & \mathbf{u}^{inf} \leq \mathbf{u} \leq \mathbf{u}^{sup} \\ & \mathbf{y}^{inf} \leq \mathbf{y}^{p} \leq \mathbf{y}^{sup} \end{aligned} \tag{4.8}$$

Further, Equation (4.8) becomes (4.9), if the optimization problem with linear predictive model presents no constraints.

$$\begin{aligned} \mathbf{u}^{\star} = \quad & \underset{\mathbf{u}}{\operatorname{argmin}}\, J\left(\mathbf{u}, \mathbf{y}^{p}, \mathbf{y}^{sp}\right) \\ \text{s.t.} \quad & \mathbf{x}^{k+1} = A\mathbf{x}^{k} + B\mathbf{u}^{k} \\ & \mathbf{y}^{p} = C\mathbf{x}^{k+1} \end{aligned} \tag{4.9}$$

4.4 Jaya-MPC Algorithm

Algorithm 2 shows the pseudo-code of the first version — with fixed weight factors $(\mathrm{r}_1, \mathrm{r}_2)$ — of the proposed Jaya-MPC algorithm; it highlights in red the differences from the original Jaya algorithm (see Algorithm 1). These differences can be summarized as follows:

- Jaya-MPC uses fixed weight factors instead of the original Jaya algorithm that uses random variables with uniform distribution as the weight factor. This modification gives Jaya-MPC deterministic behavior to avoid large steps [6] when calculating the new individuals since the feasible search-space in the power converters applications ranges from -1 to 1, i.e., the feasible values for the modulation index;

- The initial population set has a small size ($\mathrm{M} = 3$)[7] and is defined based on the upper and lower limits of the system's input ($\mathbf{u}$); the original Jaya algorithm generates initial population randomly;

- Inside the loop of the Jaya algorithm, it includes the function that evaluates the prediction over the population;

- The stop criteria in the original Jaya algorithm is only the maximum number of generations, which is a parameter the user defines. In the Jaya-MPC, besides this parameter, the user sets a tolerance (*tol*) for the cost function since it is a regulation problem. Hence, the algorithm stops and returns the optimal[8] control effort if it achieves a value less than the tolerance.

[6] The size of the deterministic weight factors and the analysis of different weighting strategies proposed in this work are discussed in Chapter 5.

[7] $\mathrm{M} = 3$ is the smallest population size allowed in the Jaya algorithm, which reduces the algorithm's computational cost.

[8] The optimization literature often uses the term "optimal" to describe the solution of the optimization algorithm given the tolerance or other stopping criteria. Almost all optimization algorithms do not return the exact value of the local or global optimum, but if it returns a solution within the tolerance criteria, this solution is called the "optimal" control effort.

Algorithm 2: Proposed Jaya-MPC algorithm.

Data: cost function: $J : \mathbb{R}^n \rightarrow \mathbb{R}$
output reference: $\mathbf{y}^{sp}$
input constraints: $\{\mathbf{u}^{inf}, \mathbf{u}^{sup}\}$
output constraints: $\{\mathbf{y}^{inf}, \mathbf{y}^{sup}\}$

Result: optimal solution: $\mathbf{u}^{\star} = \underset{\mathbf{u}}{\text{argmin}}\, J(\mathbf{u}, \mathbf{y}^p)$
optimal: cost $\mathrm{J}^{\star} = min\left(J(\mathbf{u}, \mathbf{y}^p)\right)$

initialize:
    number of generations (N)
    population size (M = 3)
    fixed weight factors[9] $(\mathrm{r}_1, \mathrm{r}_2)$
    initial population set $\left(\mathbf{U}^o = \left\{\mathbf{u}^{inf}, \frac{\mathbf{u}^{sup} - \mathbf{u}^{inf}}{2}, \mathbf{u}^{sup}\right\}\right)$
begin
    foreach *generation:* $n \leq \mathrm{N}$ **do**
        foreach $\mathbf{u}_{\mathrm{n,m}} \in \mathbf{U}^n$ **do**
            evaluate predictions: $\mathbf{y}^p = F\left(\mathbf{x}, \mathbf{u}\right)$
            calculate the cost function: $J\left(\mathbf{y}^p, \mathbf{u}, \mathbf{y}^{sp}\right)$
            select best and worst solution: $(\mathbf{u_b}, \mathbf{u_w})$
            if $J(\mathbf{u_b}) < \mathrm{J}^{\star}$ **then**
                $\mathrm{J}^{\star} = J(\mathbf{u_b})$
                $\mathbf{u}^{\star} = \mathbf{u_b}$
            end
            evaluate new individual: $(\mathbf{u}_{\mathrm{n+1,m}})$
                $\mathbf{u}_{\mathrm{n+1,m}} = \mathbf{u}_{\mathrm{n,m}} + \mathrm{r}_1\left(\mathbf{u_b} - |\mathbf{u}_{\mathrm{n,m}}|\right) - \mathrm{r}_2\left(\mathbf{u_w} - |\mathbf{u}_{\mathrm{n,m}}|\right)$
        end
        if $\mathrm{J}^{\star} < tol$ **then**
            return $(\mathbf{u}^{\star}, \mathrm{J}^{\star})$
        end
    end
    return $(\mathbf{u}^{\star}, \mathrm{J}^{\star})$
end

[9] The first version of this algorithm uses constant values for the weight factors. However, an adaptive approach that evaluates r_1 and r_2 in each generation will be discussed in the next chapter.

Table 4.1 describes the differences between the proposed Jaya-MPC and the original Jaya algorithm (see Algorithm 1), which were highlighted in red in Figure 4.1.

Table 4.1: Comparison between the original Jaya algorithm (see Algorithm 1) and the proposed Jaya-MPC algorithm (see Algorithm 2).

	original Jaya algorithm (1)	proposed Jaya-MPC (2)
initial population	random and large	deterministic and small
weight factors	random	deterministic
stop criteria	fixed generations	cost tolerance

4.4.1 Initial Population

The initial population of the algorithm is a set of discrete control actions $\mathbf{U}^o = \{\mathbf{u}_{0,1}, \ldots, \mathbf{u}_{0,\mathrm{M}}\}$ containing only three individuals ($\mathrm{M} = 3$) — the minimum set size the algorithm needs to work [10] — to reduce the computational cost of the algorithm.

Hence, Jaya-MPC initial population can be defined as:

$$\mathbf{U}^o = \left\{ \mathbf{u}^{inf}, \frac{\mathbf{u}^{sup} - \mathbf{u}^{inf}}{2}, \mathbf{u}^{sup} \right\} \tag{4.10}$$

Where $\mathbf{u}^{sup}$ and $\mathbf{u}^{inf}$ are the maximum and minimum inputs allowed in the system, i.e., the maximum and minimum modulation indexes allowed in the converter.

From the initial population — composed of the three levels of the power converter —, the algorithm progresses and calculates new individuals, based on the rule of Equation 4.11, that makes each individual of the population move away from the worst solution ($\mathbf{u_w}$) and approaches the best solution ($\mathbf{u_b}$).

$$\mathbf{u}_{\mathrm{n+1,m}} = \mathbf{u}_{\mathrm{n,m}} + \mathrm{r}_1 \left(\mathbf{u_b} - |\mathbf{u}_{\mathrm{n,m}}|\right) - \mathrm{r}_2 \left(\mathbf{u_w} - |\mathbf{u}_{\mathrm{n,m}}|\right) \tag{4.11}$$

It means that the algorithm finds a continuous set of control efforts from a discrete set — as a contrast to the common FCS-MPC approaches, which evaluate only discrete actions: the converter's switching states. After solving the optimization problem, the algorithm returns the optimal solution ($\mathbf{u}^\star$), a set of continuous signals, i.e., the modulation indexes, and provides them to a PWM block to generate switching pulses with fixed switching frequency.

This change of paradigm, which is the core of GPC, makes the number of predictions the algorithm assesses independent of the quantity of switching states of

[10] The Jaya algorithm selects the best and the worst solution among those in the current population in every step. It means that the algorithm needs more than two individuals in the population.

the power converter. Thus, the computational cost — directly associated with the number of predictions — of the proposed technique is unaffected by the power converter topology and depends only on the optimization algorithm, the filter model, and the cost function.

This topology independence feature could lead to a practical benefit in applications such as multilevel converters, which present a large set of switching states. For instance, in a typical FCS-MPC, this set would feed the predictor block, leading to an increase in the total number of predictions — even when switching-states reduction techniques are employed [110]. Hence, this prediction increment leads to more cost function evaluations, which raises the computational cost. Although this application is beyond the scope of this thesis, it may motivate future work.

Although the classical FCS-MPC is known for the lowest computational cost among MPC techniques for typical two-level converters, it often demands high sampling frequencies to compensate for variable-switching frequency effects, such as high ripples and zero-crossing issues. The analyses of chapter 6 address this trade-off, which is a key element in validating the Jaya-MPC as a viable solution with low computational cost and fixed switching frequency.

4.4.2 Stop Criteria

The original Jaya algorithm evaluates a new population until it reaches a defined number of generations (N). Then, the algorithm returns an approximation of the optimal solution.

Jaya-MPC still sets a maximum number of generations, but as the control objective is a regulation problem, the minimum cost is zero. So, if the cost function value of the best solution ($\mathbf{u_b}$) of the current generation is lower than a defined tolerance (tol), the algorithm stops and returns the optimal modulation index ($\mathbf{u}^\star$).

In this work, the tolerance error in the regulation problem is equal to 1 %. Therefore, as the cost function is quadratic, the tolerance is 1×10^{-4}, i.e., if the cost function value for a determined solution is less than this tolerance ($\mathrm{J}^\star < 1 \times 10^{-4}$), the regulation error is less than 1%.

4.4.3 Weighting Factors Strategy

In the original Jaya algorithm, the weighting factors r_1 and r_2 are random variables (see Algorithm 1). This approach leads to non-deterministic behavior, which may be undesired in the applications of this work. Hence, it is necessary to define deterministic values for r_1 and r_2.

These values play a crucial role in the performance of the Jaya-MPC. If they are too large, the new population maybe not be inside the feasible control-effort limits $[-1, 1]$; if they are too small, the algorithm would need several iterations to reach the optimal solution.

This work proposes two approaches to address this problem: fixed-weights strategy and adaptive-weights strategy; these strategies' goals are to reduce the total number of generations of the algorithm and enhance the controller's performance.

The fixed-weights strategy sets constant values for both weight-factors (r_1, r_2), while in the adaptive-weights approach, it sets initial values for r_1 and r_2 and, as the algorithm proceeds, it recalculates both weight-factors in each generation, according to Equation (4.12). The next chapter details the analysis of the performance of both approaches.

$$(r_1, r_2) = \left(\frac{r_1}{n}, \frac{r_2}{n}\right) \tag{4.12}$$

The fixed-weights approach fits in the proposed Jaya-MPC algorithm (see Algorithm 2). However, the adaptive-weights strategy demands an alteration, resulting in the following adaptive Jaya-MPC Algorithm (3).

The only difference (highlighted in red) to the Jaya-MPC algorithm is the inclusion of the procedure that recalculates the weight-factors inside the generation loop, according to Equation (4.12).

Algorithm 3: Jaya-MPC algorithm with adaptive weight-factors strategy.

Data: cost function: $J : \mathbb{R}^n \rightarrow \mathbb{R}$
output reference: $\mathbf{y}^{sp}$
input constraints: $\{\mathbf{u}^{inf}, \mathbf{u}^{sup}\}$
output constraints: $\{\mathbf{y}^{inf}, \mathbf{y}^{sup}\}$

Result: optimal solution: $\mathbf{u}^{\star} = \underset{\mathbf{u}}{\text{argmin}}\, J(\mathbf{u}, \mathbf{y}^p)$
optimal: cost $\mathrm{J}^{\star} = min\left(J(\mathbf{u}, \mathbf{y}^p)\right)$

initialize:
  number of generations (N)
  population size ($\mathrm{M} = 3$)
  initial weight factors $(\mathrm{r}_1, \mathrm{r}_2)$
  initial population set $\left(\mathbf{U}^o = \left\{\mathbf{u}^{inf}, \frac{\mathbf{u}^{sup} - \mathbf{u}^{inf}}{2}, \mathbf{u}^{sup}\right\}\right)$
begin
  foreach *generation:* $n \leq \mathrm{N}$ **do**
    evaluate adaptive weight-factors: $(\mathrm{r}_1, \mathrm{r}_2) = \left(\frac{\mathrm{r}_1}{n}, \frac{\mathrm{r}_2}{n}\right)$
    foreach $\mathbf{u}_{\mathrm{n,m}} \in \mathbf{U}^n$ **do**
      evaluate predictions: $\mathbf{y}^p = F(\mathbf{x}, \mathbf{u})$
      calculate the cost function: $J(\mathbf{y}^p, \mathbf{u}, \mathbf{y}^{sp})$
      select best and worst solution: $(\mathbf{u_b}, \mathbf{u_w})$
      if $J(\mathbf{u_b}) < \mathrm{J}^{\star}$ **then**
        $\mathrm{J}^{\star} = J(\mathbf{u_b})$
        $\mathbf{u}^{\star} = \mathbf{u_b}$
      end
      evaluate new individual: $(\mathbf{u}_{\mathrm{n+1,m}})$
        $\mathbf{u}_{\mathrm{n+1,m}} = \mathbf{u}_{\mathrm{n,m}} + \mathrm{r}_1\left(\mathbf{u_b} - |\mathbf{u}_{\mathrm{n,m}}|\right) - \mathrm{r}_2\left(\mathbf{u_w} - |\mathbf{u}_{\mathrm{n,m}}|\right)$
    end
    if $\mathrm{J}^{\star} < tol$ **then**
      return $(\mathbf{u}^{\star}, \mathrm{J}^{\star})$
    end
  end
  return $(\mathbf{u}^{\star}, \mathrm{J}^{\star})$
end

4.5 Jaya-MPC For Ac Current Control

This work proposes using the Jaya-MPC algorithm for ac-current control in the $\alpha\beta$-frame of a three-phase converter connected to the grid since this scope is the most common MPC application in power electronics.

Figure 4.2 depicts the block diagram of the system studied in this work: the plant is a three-phase dc-ac converter, with an inductive filter connected to the grid. A pq-theory block generates the current references ($i^{sp}_{L_{\alpha\beta}}$) based on the active (p^{sp}) and reactive power references (q^{sp}).

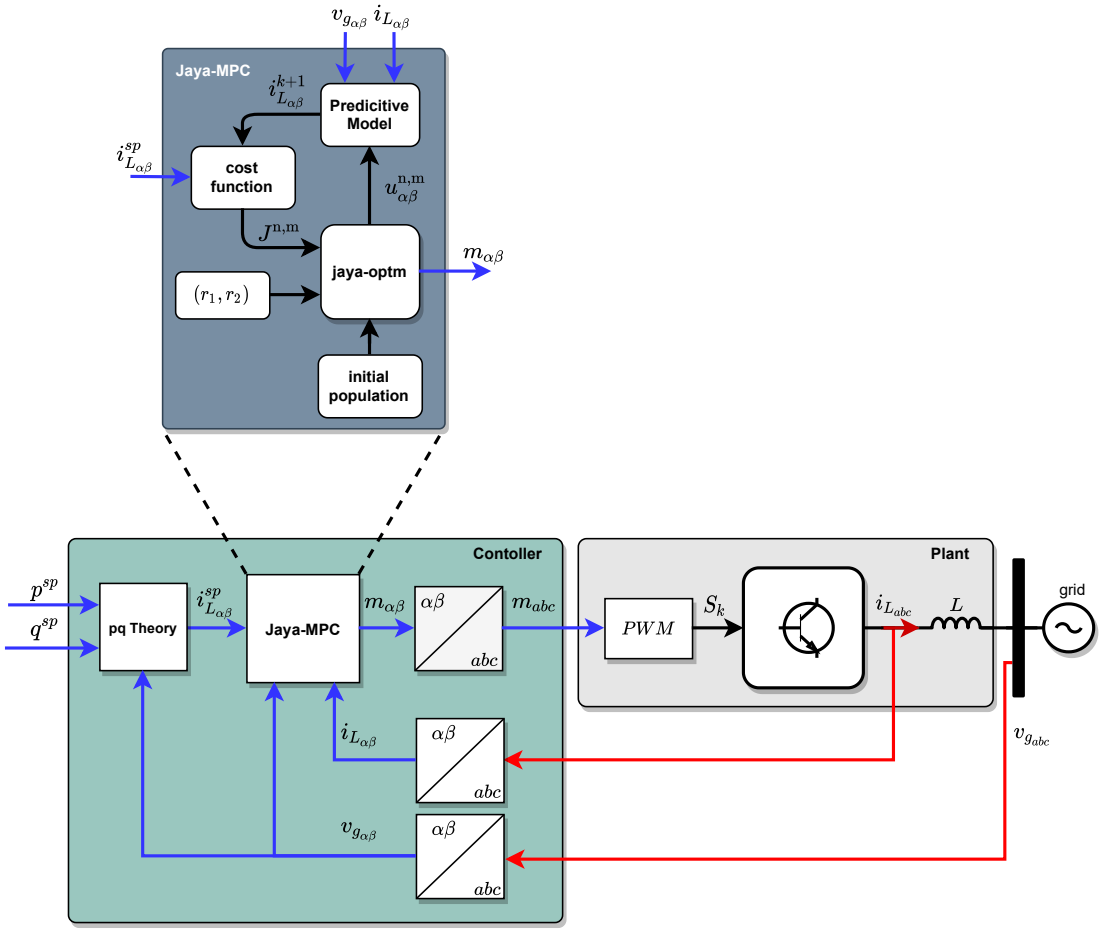


Figure 4.2: Block diagram of the Jaya-MPC ac-current control loop in the $\alpha\beta$-frame.

The system inputs $\mathbf{u}$ are the modulation indexes ($m_{\alpha\beta}$), while the systems outputs are the ac-currents ($i^{sp}_{L_{\alpha\beta}}$), both referred to the $\alpha\beta$-frame. Hence, the initial population is a set of modulation indexes with a superior and inferior bounds equal to 1 and -1, respectively, which leads to expressions of Equations (4.13) and (4.13).

$$\mathbf{U}^o_\alpha = \{-1, 0, 1\} \tag{4.13}$$

$$\mathbf{U}^o_\beta = \{-1, 0, 1\} \tag{4.14}$$

These Equations mean that the Jaya-MPC algorithm starts from an initial population containing the upper and lower limits of the control effort and progresses to converge to the optimal modulation index, given the cost function of Equation (4.15), which contains both quadratic error terms and the penalty term that penalizes overcurrent values.

$$J = ||i^{sp}_{L_{\alpha\beta}} - i^{p}_{L_{\alpha\beta}}||^2_2 + P_y(i_{L_{\alpha\beta}}) + P_u(m_{\alpha\beta}) \tag{4.15}$$

4.5.1 Predictive Model

For the three-phase converter presented in Figure 4.2,the MPC strategies studied in this work uses Equation (4.16) and (4.17) to predict the dynamics of the ac currents in the $\alpha\beta$-frame.

The discretization process of the system's state-equation used the Forward-Euler method, which is the most common approach to derive the predictor model for power converter applications [51–53].

$$i^{p}_{L_\alpha} = \left(1 - \frac{R\,T_s}{L}\right) i^{k}_{L_\alpha} + \frac{T_s}{L} v^{k}_{g\,\alpha} - m^{k}_{\alpha} \frac{T_s v^{k}_{dc}}{2L} \tag{4.16}$$

$$i^{p}_{L_\beta} = \left(1 - \frac{R\,T_s}{L}\right) i^{k}_{L_\beta} + \frac{T_s}{L} v^{k}_{g\,\beta} - m^{k}_{\beta} \frac{T_s v^{k}_{dc}}{2L} \tag{4.17}$$

R and L are the filter's resistance and inductance; T_s is the sampling period; and $v^k_{g_{\alpha\beta}}$, $i^k_{L_{\alpha\beta}}$, and v^k_{dc} are the measurements, at the sampling-instant k, of the grid voltage, the filter's ac-current, and the dc-link voltage, respectively.

Chapter 5

Parametric Analysis

The Jaya-MPC algorithm uses deterministic values as weighting parameters (r_1, r_2) in contrast to the original Jaya algorithm that uses random variables. This difference motivates the study of two strategies to set weight parameters of Jaya-MPC: fixed weights and adaptive weights. Besides, this study also considers the relation between r_1 and r_2, i.e., if these values are correlated or not, and what is the correlation rule that fits best the goal of improving steady-state performance while reducing the average number of generations that Jaya-MPC requires to achieve the optimal control-effort, given the cost function. This chapter covers four analyses regarding the weighting-factor strategy of the Jaya-MPC algorithm and the correlation between both weight-factors (r_1, r_2):

- Non-correlated fixed-weights strategy analysis;
- Non-correlated adaptive-weights strategy analysis;
- Correlated fixed-weights strategy analysis;
- Correlated adaptive-weights strategy analysis.

5.1 Simulation Setup

These analyses are carried out in $MATLAB^{TM}/Simulink^{TM}$, which runs a parametric sweep on Jaya-MPC; for each combination of the parameters r_1 and r_2, the procedures evaluate the values of three metrics:

- Average number-of-generations until Jaya-MPC reaches the optimal solution, given a tolerance;
- Average optimal-cost that Jaya-MPC achieves;
- Total-harmonic-distortion (THD) of the ac-currents.

Figure 5.1 shows the flowchart of the $MATLAB^{TM}$ algorithm that runs the parametric analyses; for each analysis, it sets the simulation parameters (according to Table 5.1), and runs the $Simulink^{TM}$ system, for each pair of weight-factors (r_1, r_2) in the parametric sweep.

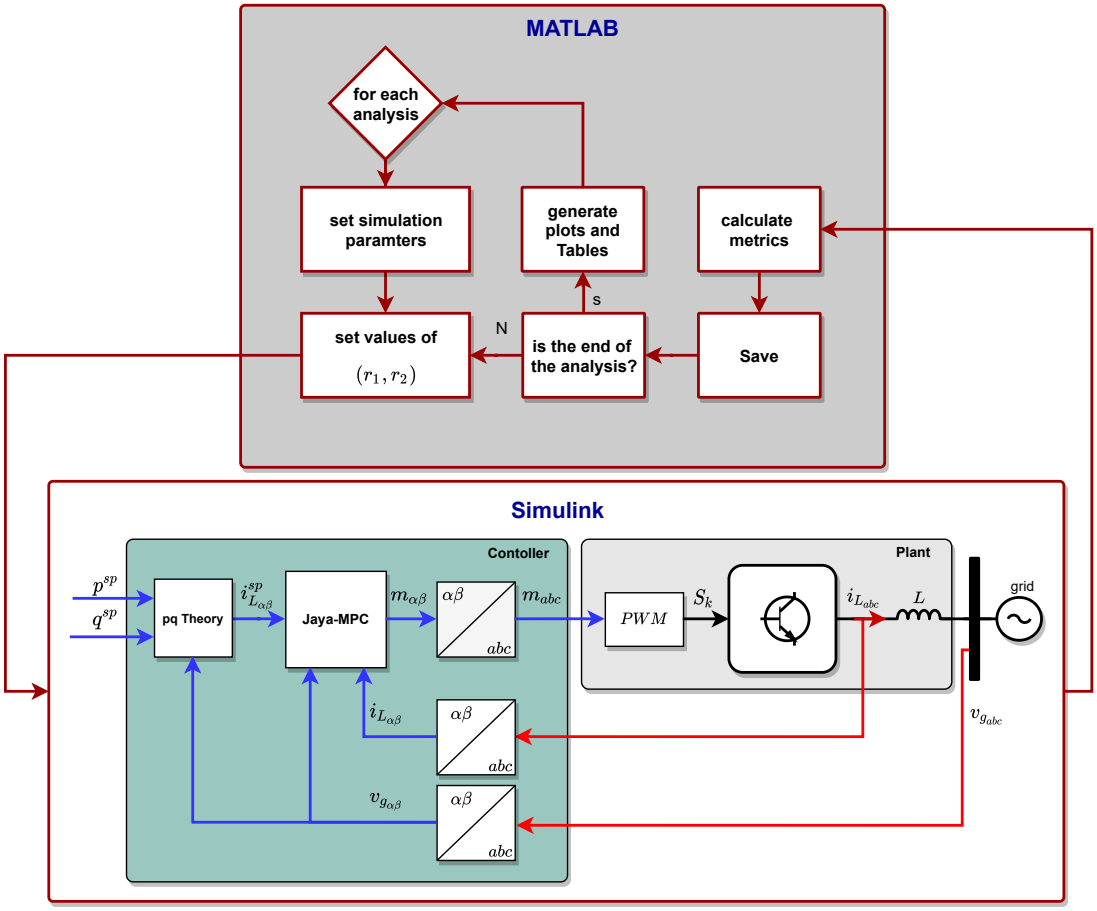


Figure 5.1: Flowchart of the $MATLAB^{TM}/Simulink^{TM}$ simulation procedure, which automatically runs the parametric analyses regarding the weighting-factors strategies.

After each simulation, the $MATLAB^{TM}$ algorithms evaluate the three metrics, save the variables, and, when the analysis ends, generate the plots and tables. The modeled system comprises a three-phase converter connected to an ideal grid with Jaya-MPC block controlling the ac-current, given a step in the power references. The metrics are assessed over two periods of the fundamental frequency after the system achieves steady-state.

Table 5.1: Simulation parameters for the parametric analyses.

Nominal Electrical Parameters	
$V_{rms,LL}$	220 V
V_{dc}	450 V
S	10 KVA
f	60 Hz
Filter Parameters	
L	2.03 mH
R	30.6 mΩ
Jaya-MPC Parameters	
N	8
tol	10^{-4}
Switching Frequency (F_{sw})	5940 Hz
Sampling Period ($\frac{1}{F_{sw}}$)	168.35 μs
Simulation Parameters	
simulation time step	1.68 μs
simulation time	0.0667 s (4 cycles)

5.2 Non-Correlated Fixed-Weights Strategy

The non-correlated fixed-weights strategy is the first approach to answer the question: is there any combination of fixed values for r_1 and r_2 that could reduce the average number of generations while maintaining a suitable steady-state performance?

The analysis of this section runs a 2-dimensional parametric sweep for each combination of r_1 and r_2, with no correlation between them, i.e., both are independent values. The limits of both weight factors range from 0.05 to 1.0.

5.2.1 Analysis of The Average Number of Generations

Figure 5.2 depicts the parametric-sweep result of the average number of generations that Jaya-MPC takes to return the optimal control-effort — a measurement of how the parameters r_1 and r_2 affect the performances of the algorithm.

In (a), it presents the parametric surface of the average number of generations produced by the parametric analysis; in (b), it depicts the heat map, i.e., the top-view of (a). These data show a rough surface with no correlation between r_1 and r_2; Besides, none of the sets (r_1, r_2) can reach an average number of generations below four.

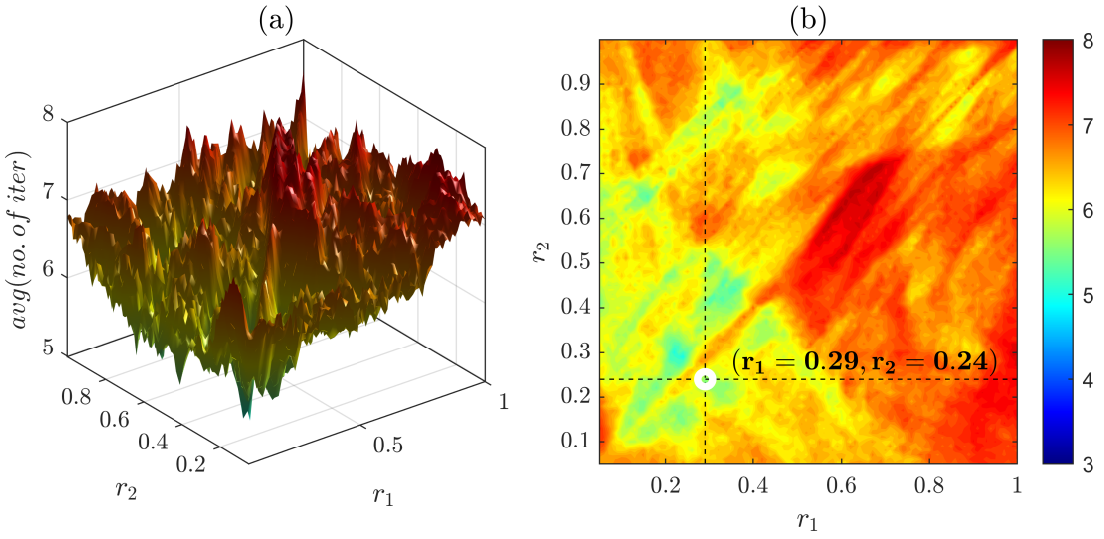


Figure 5.2: Non-correlated parametric analysis of the influence of Jaya weight parameters (r_1, r_2) in the performance of Jaya-MPC: (a) 3D-view and (b) 2D-view of the time-average number of generations. Lower average number of generations (4.99) achieved by using $(r_1 = 0.29, r_2 = 0.24)$.

5.2.2 Analysis of The Average Optimal Cost

Figure 5.3 depicts the results of the parametric-sweep for the average optimal-cost that Jaya-MPC achieves in steady-state.

In (a), it presents the parametric surface of the average number of generations produced by the parametric analysis; in (b), it depicts the heat map, i.e., the top-view of (a). Jaya-MPC reaches the lowest average optimal cost using lower values of $r1$ and $r2$ in the left lower corner of (b); the other corners present higher costs, which means that the regulation of the ac-currents in these corners are worse than the one in the left lower corner. These results indicate a relation of symmetry between r_1 and r_2 — around the line $r_1 = r_2$ —, which is the first clue that it may be a correlation rule between both parameters to be studied and explored.

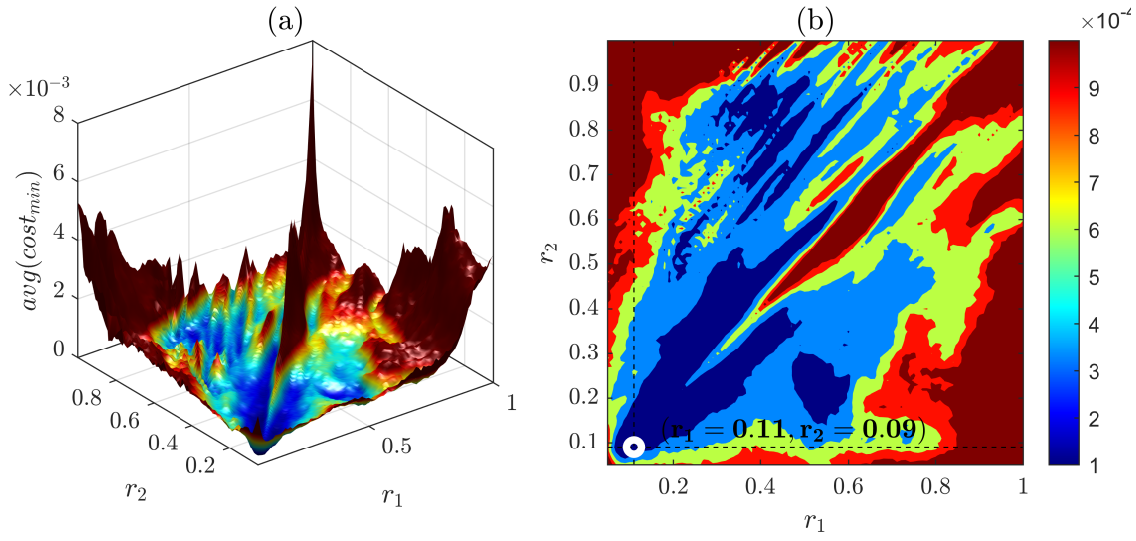


Figure 5.3: Non-correlated parametric analysis of the influence of Jaya weight parameters (r_1, r_2) in the performance of Jaya-MPC: (a) 3D-view and (b) 2D-view of the time-average cost of ac current control performed by Jaya-MPC. Lower average optimal-cost (5.59×10^{-5}) achieved by using $(r_1 = 0.11, r_2 = 0.09)$.

5.2.3 Analysis of The THD

Figure 5.4 shows the results of the parametric sweep for the THD of the ac-currents. In (a), it presents the parametric surface of the THD for each set of r_1 and r_2; in (b), it depicts the heat map, i.e., the top-view of (a).

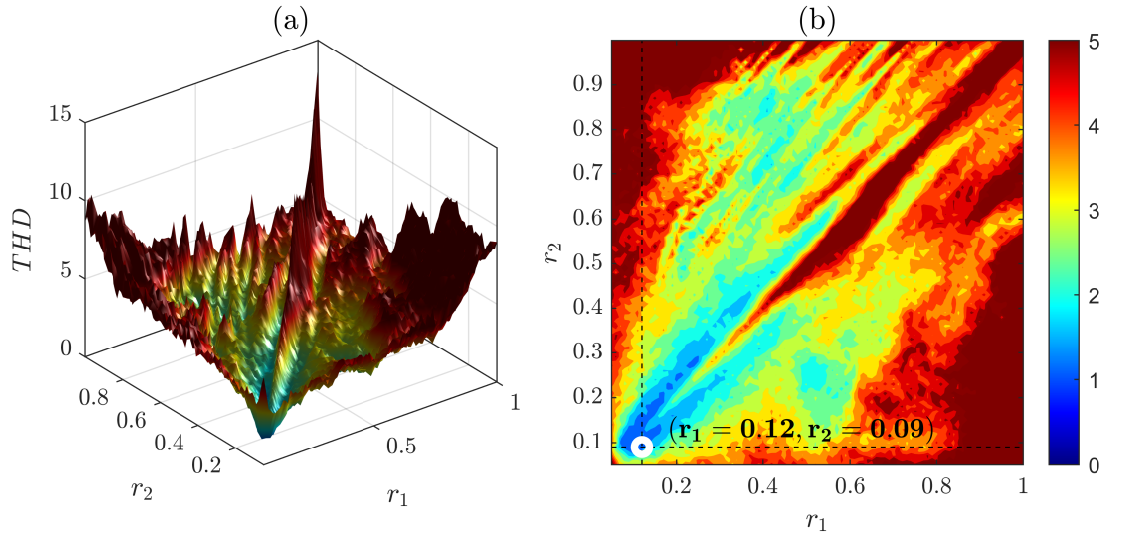


Figure 5.4: Non-correlated parametric analysis of the influence of Jaya weight parameters (r_1, r_2) in the performance of Jaya-MPC:(a) 3D-view and (b) 2D-view of the THD (%). Lower THD(1.1%) achieved by using $(r_1 = 0.12, r_2 = 0.09)$.

The surfaces of Figures 5.4 and Figure 5.3 present similar shapes because both THD and average-optimal cost strongly correlate. An explanation for that is that the cost function is a current regulation, and the better the regulation performance, the lower the THD. Hence, this metric presents the same correlation rule observed in Figure 5.3 (b).

5.3 Non-Correlated Adaptive-Weights Strategy

The non-correlated adaptive-weights strategy considers that, as the Jaya-MPC calculates new generations and approaches the optimal value, the steps to calculate the next individuals of the population may be smaller since they are closer to the optimal solution.

In other words, this strategy decreases the values of the weights r_1 and r_2 (see Equation (4.12)) as the number of generations increases, allowing the algorithm to reach the optimal solution in a more precise way and reducing the average number of generations spent by the algorithm.

The analysis of this section runs a 2-dimensional parametric sweep on Jaya-MPC and computes the values of the same three metrics for each combination of r_1 and r_2, with no correlation between them.

5.3.1 Analysis of The Average Number of Generations

Figure 5.5 depicts the parametric-sweep result of the average number of generations that Jaya-MPC takes to return the optimal control-effort — a measurement of how the adaptive parameters r_1 and r_2 affect the performances of the algorithm.

In (a), it presents the parametric surface of the average number of generations produced by the parametric analysis; in (b), it depicts the heat map, i.e., the top-view of (a). The parameters r_1 and r_2, shown in (a) and (b), are the initial ones, i.e., Jaya-MPC starts running these pair of weights, and as the generations increase, it updates the values of r_1 and r_2 using Equation (4.12).

A comparison with the equivalent analysis of Figure 5.2 shows that the use of the adaptive-weights strategy leads to a reduction in the average number of generations to reach the optimal control effort — note that the heat map scale of both Figures 5.5 and 5.2 are the same and now Jaya-MPC achieves an average number of generations below 4.

Instead of the equivalent analysis of Figure 5.2, the adaptive-weights strategy presents a correlation region between r_1 and r_2 that can be explored: two well-defined valley in the neighborhood of the region $r_1 = r_2$ — one, with a local-minima, in the top-right corner of (b), and the other centered at the point $(r_1 = 0.3, r_2 = 0.28)$.

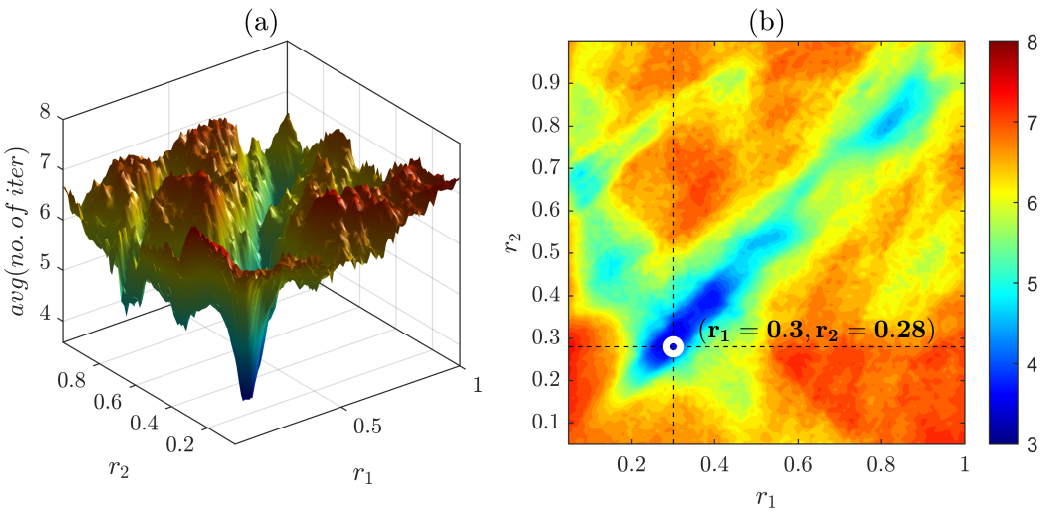


Figure 5.5: Non-correlated parametric analysis of the influence of the adaptive-weight parameters $(\mathrm{r}_1, \mathrm{r}_2)$ in the performance of Jaya-MPC: (a) 3D-view and (b) 2D-view of the time-average number of generations. Lower average number of generations (3.58) achieved by using $(\mathrm{r}_1 = 0.3, \mathrm{r}_2 = 0.28)$.

5.3.2 Analysis of The Average Optimal Cost

Figure 5.6 shows that the use of the adaptive-weights strategy leads to a parametric-surface smother than one of the equivalent analyses with the fixed-weights approach (see Figure 5.3).

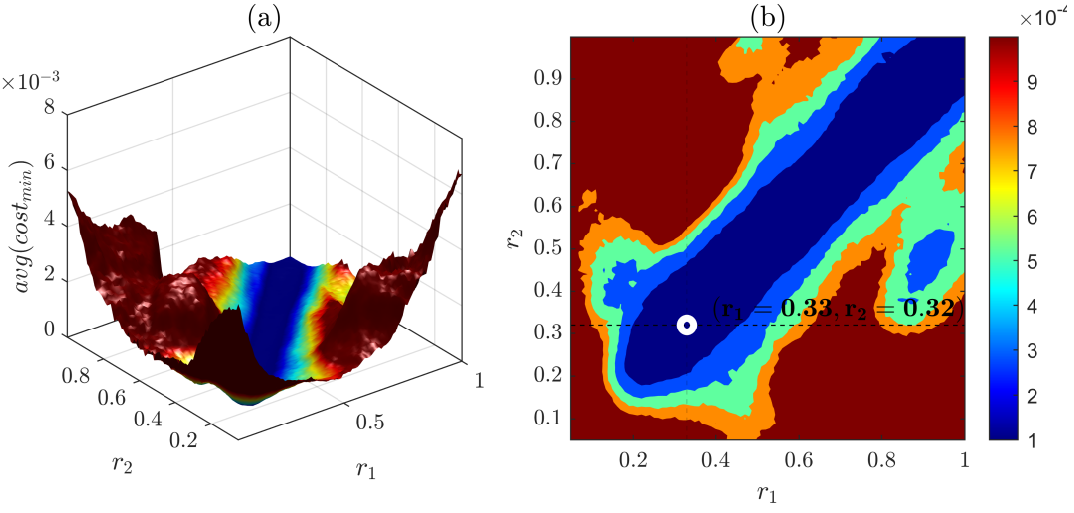


Figure 5.6: Non-correlated parametric analysis of the influence of the adaptive-weight parameters $(\mathrm{r}_1, \mathrm{r}_2)$ in the performance of Jaya-MPC: (a) 3D-view and (b) 2D-view of the average optimal-cost; Lower average cost (3.18×10^{-5}) is achieved by using $(\mathrm{r}_1 = 0.33, \mathrm{r}_2 = 0.32)$.

In other words, while the fixed-weights strategy presents several local valleys, with only a limited region with a lower average cost, this strategy results in a surface with one well-defined valley around the line defined by $r_1 = r_2$. This finding suggests that the adaptive-weights strategy is a more robust approach since it achieves lower cost function values for a wide range of the parameters r_1 and r_2.

5.3.3 Analysis of The THD

Figure 5.7 shows the parametric surfaces of the THD value of the ac-currents when Jaya-MPC uses the adaptive-weights strategy. The results show that this strategy produces a surface with a valley around the region $r_1 = r_2$, in contrast with the fixed-weighs strategy (see Figure 5.4) that produces a surface with several valleys. This feature is a valuable asset of the adaptive-weights strategy since it results in a well-defined region where Jaya-MPC reaches low THD values, which means a more robust approach to set the parameters r_1 and r_2.

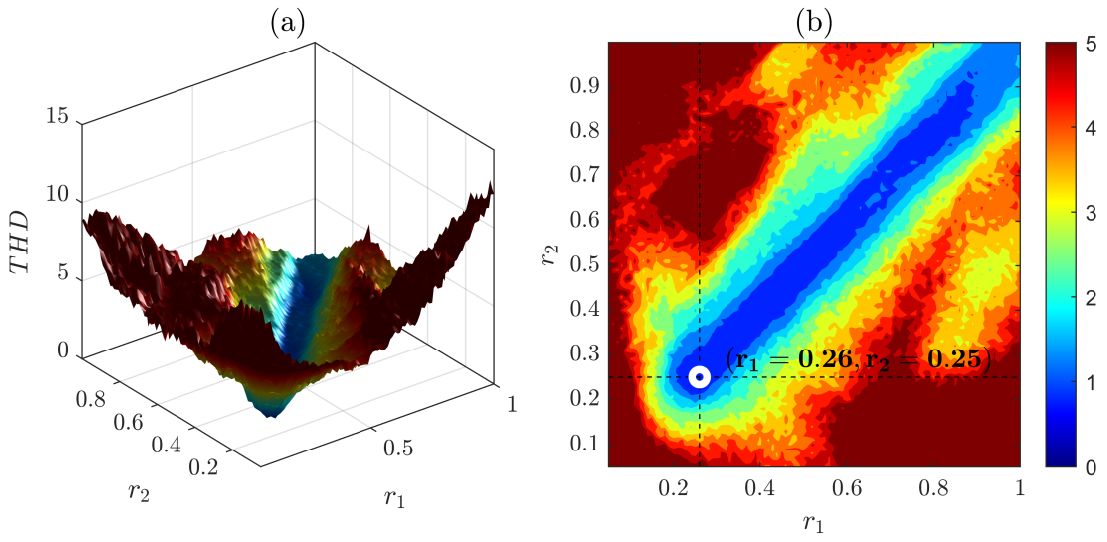


Figure 5.7: Non-correlated parametric analysis of the influence of the adaptive-weight parameters (r_1, r_2) in the performance of Jaya-MPC: (a) 3D-view and (b) 2D-view of the THD (%) of the ac-currents. Lower THD(0.8%) achieved by using $(r_1 = 0.26, r_2 = 0.25)$.

5.4 Correlated-Weights Strategy

The non-correlated parametric analyses show that both fixed-weights and adaptive-weights strategies present symmetric behavior around the line $r_1 = r_2$.

However, the fixed-weights strategy does not produce a well-defined valley in the region $r_1 = r_2$, as the surfaces of each metric are very sharp. On the other hand,

in the case of the adaptive-weights strategy, the analyses show a well-defined valley around the line $r_1 = r_2$, resulting in a robust region from the point-of-view of the parameter setting.

These findings lead to a correlation between the parameters r_1 and r_2. Thus, it is only necessary to design one parameter in the Jaya-MPC, making it simpler to set and analyze.

Figure 5.8 shows the results of the correlation-rule ($r_1 = r_2$) in both fixed-weights and adaptive-weights strategies: (a) depicts the average number of generations that Jaya-MPC takes to return the optimal modulation index to control the ac currents; (b) shows the average optimal-cost achieved by the Jaya-MPC algorithm; (c) presents THD of the ac currents.

These results lead to the following findings:

- Although the fixed-weights strategy can achieve values of THD and average optimal cost close to the ones produced by the adaptive-weights strategy, it takes, on average, more generations than the adaptive-weights strategy — this finding dismisses the fixed-weight strategy;

- For values of r_1 greater than .25, the adaptive-weights strategy results in an almost-constant low THD (less than 2.0 %) and an almost-constant average cost less than the tolerance ($tol = 1 \times 10^{-4}$), which means a robust and reliable strategy — this finding confirm that the correlated adaptive-weights strategy is a simpler and more reliable solution.

Table 5.2 presents the values of each metric for three possible configuration of the parameters r_1 and r_2 using the correlation rule $r_1 = r_2$. These results show that both $r_1 = 0.25$ and $r_1 = 0.33$ lead to the best results: the former gives the best average minimum cost and THD, and the latter gives the lower average number of generations.

Table 5.2: Steady-state metrics of the of Jaya-MPC in $\alpha\beta$-frame with correlated adaptive-weights strategy ($r_1 = r_2$).

	$r1 = 0.25$ $r2 = 0.25$	$r1 = 0.33$ $r2 = 0.33$	$r1 = 0.5$ $r2 = 0.5$
mean min cost	3.7e-05	3.4e-05	3.9e-05
mean no. of iterations	3.98	3.78	4.94
THD(%)	0.93	0.88	0.99

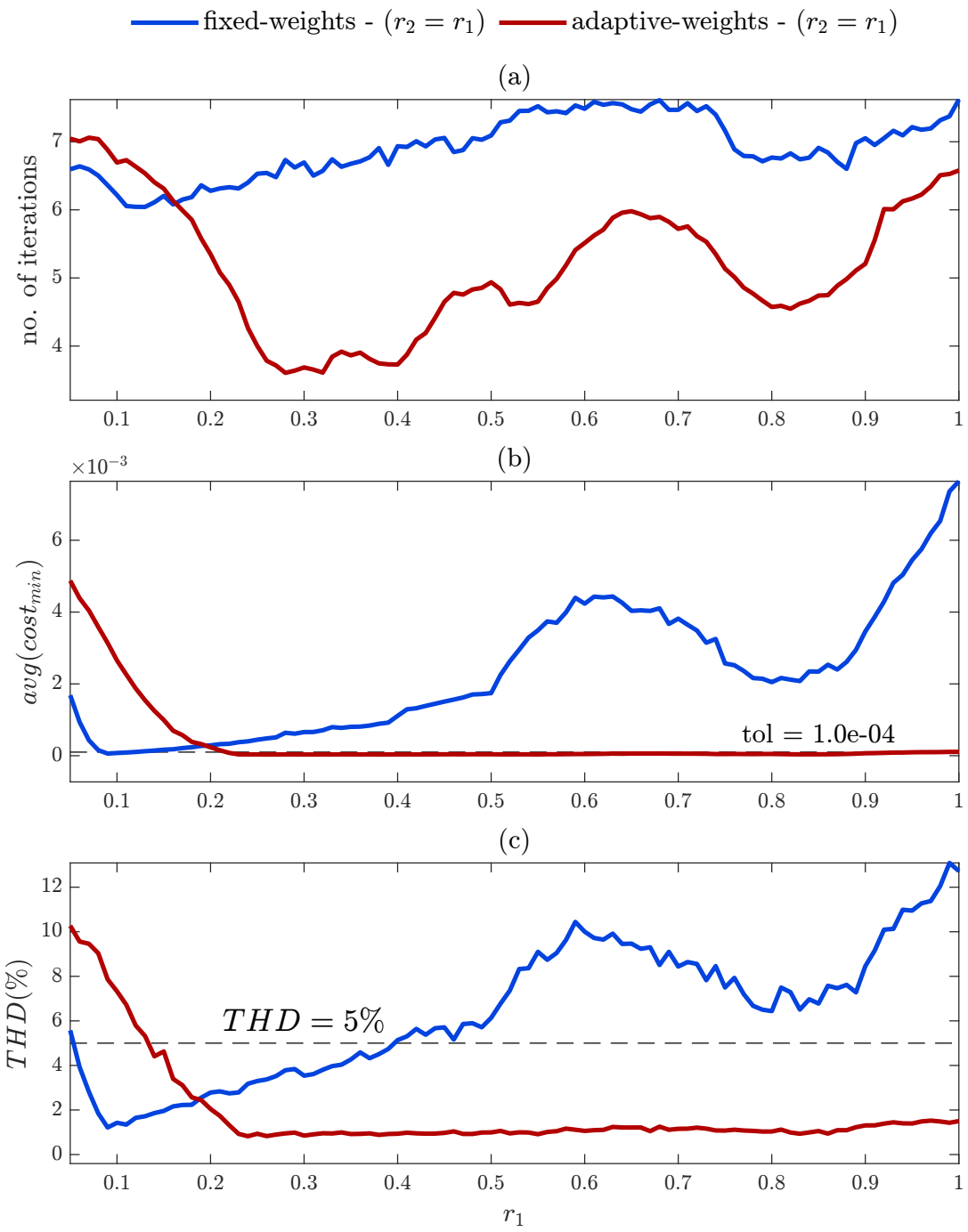


Figure 5.8: Correlated parametric analysis ($r_2 = r_1$) of the performance of Jaya-MPC using fixed-weights and adaptive-weights: (a) average number of generations to return the solution; (b) average optimum-cost; (c) ac current THD (%).

The results of this chapter support the conclusions that the adaptive-weights strategy outperforms the fixed-weights one and that the correlated rule between both weights ($r_1 = r_2$) leads to a simple design solution with only one control parameter. Thus, from now on, the following chapters use the adaptive-weights strategy with $r_1 = 0.33$ as the initial parameter of the Jaya-MPC since it gives a lower average minimum cost and THD.

Chapter 6

Switching Frequency And Computation Cost

A secondary contribution of this thesis is detailed in this chapter: a metric to evaluate the switching frequency spread of FCS-MPC. This new metric is based on an instantaneous estimation of the switching frequency. The following analyses show the difficulty of setting the sampling frequency of FCS-MPC because of its non-linear switching frequency profile that produces non-characteristic harmonics. Also, at the end of this chapter, it presents a discussion regarding the computational cost of both FCS-MPC and Jaya-MPC based on how many predictions these two strategies evaluate.

6.1 Analysis On The Switching-Frequency Spread

The work presented in [111] estimates the switching frequency of power converters using a simple algorithm, based on the time that elapses between switching state transitions, i.e., the time between positive- and negative-edge commutations.

The advantage of this algorithm is that it estimates the switching frequency instantaneously, which makes it suitable for analyzing the switching frequency spread-spectrum profile.

Figure 6.1 depicts the estimation of the switching frequency ($\hat{f}_{sw}$) of a standard PWM block using a sinusoidal modulation index with fixed amplitude and frequency (60Hz). The estimated signal presents the instantaneous switching frequency with an average value of 5940 Hz, the exact value of the switching frequency set in the PWM block, and symmetrical low deviations from this value, due to the effects of the fundamental component in the modulation process.

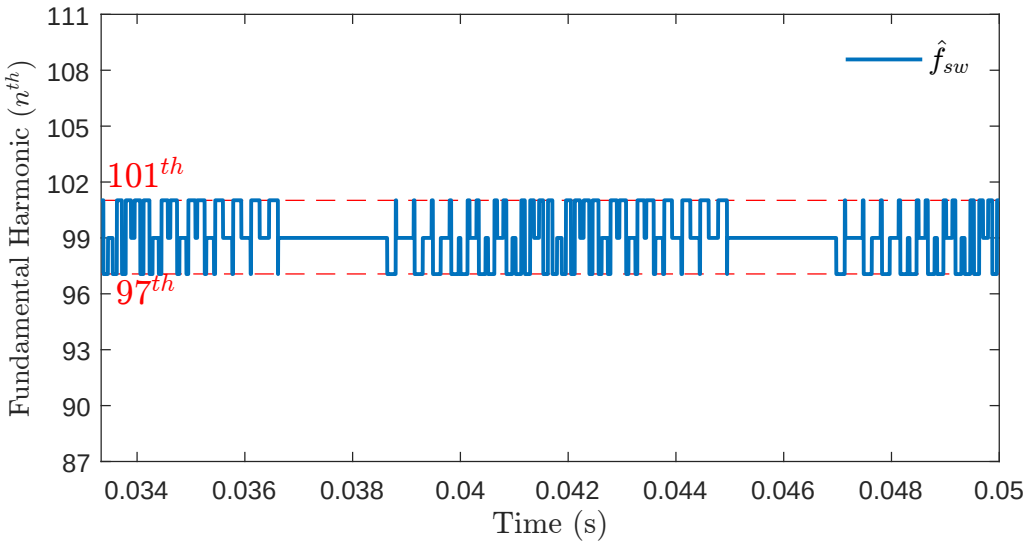


Figure 6.1: Estimation of the instantaneous switching frequency of a standard PWM module with a triangle carrier ($F_s = 5940\,Hz$) using the algorithm proposed in [111].

Based on this estimation, it is possible to extract a set of switching frequency components, $\hat{F}_{sw}$, and their respective distribution H_{sw}, i.e., the weight of each component in the switching frequency profile; Equations (6.1) and (6.2) gives these sets:

$$\hat{F}_{sw} = \{f_1^{sw}, \ldots, f_N^{sw}\} \tag{6.1}$$

$$H_{sw} = \{h_1^{sw}, \ldots, h_N^{sw}\} \tag{6.2}$$

Once having (6.1) and (6.2), one can extract the dominant frequency, $f_{sw}^{\star}$, which is the main component of $\hat{F}_{sw}$, i.e., the component with the highest weight, $h_{sw}^{\star}$, in H_{sw}. Equations (6.3) and (6.4) formalizes these steps:

$$h_{sw}^{\star} = max\{H_{sw}\} \tag{6.3}$$

$$f_{sw}^{\star} = f_i^{sw} \ , \ \{f_i^{sw} \in \hat{F}_{sw} \ / \ h_{sw}^{\star} = h_i^{sw}\} \tag{6.4}$$

6.1.1 Total Frequency Spread

This thesis proposes a metric, as a secondary contribution of this work, based on the distribution of the switching frequency estimation provided by this algorithm to quantify the total spread of the switching frequency profile of both Jaya-MPC, which uses a PWM modulator with a fixed switching frequency, and FCS-MPC, which presents a variable switching frequency, to compare them.

Equation (6.5) shows the Total-Frequency-Spread (TFS) metric. It consists of the sum of the quadratic deviations between the dominant-frequency ($f_{sw}^{\star}$) and each existing frequency (f_n^{sw}), weighted by the distribution (h_n^{sw}); this value is normalized by the dominant-frequency ($f_{sw}^{\star}$) times its respective value in the distribution profile ($h_{sw}^{\star}$).

$$TFS = \frac{1}{h_{sw}^{\star} f_{sw}^{\star}} \sqrt{\sum_{n=1}^{N} (h_n^{sw})^2 \ ||f_n^{sw} - f_{sw}^{\star}||^2} \tag{6.5}$$

6.1.2 Analysis of TFS of a PWM Module

Figure 6.2 shows that the normalized distribution of the instantaneous switching frequency (h_{sw}) of Figure 6.1 is coherent with the FFT of the switching-pulses, containing the modulator switching frequency component (99^{th}) plus the two side lobes (97^{th} and 101^{th}). Also, this outcome shows that the dominant frequency ($f_{sw}^{\star}$) of an ideal PWM-based converter equals the average switching frequency ($\bar{f}_{sw}$).

The components' amplitudes of both FFT and h_{sw} are different because h_{sw} is a normalized distribution of $\hat{f}_{sw}$, i.e., the percent time that $\hat{f}_{sw}$ is equal to each component, which is conceptually different from the meaning of the FFT component's amplitude.

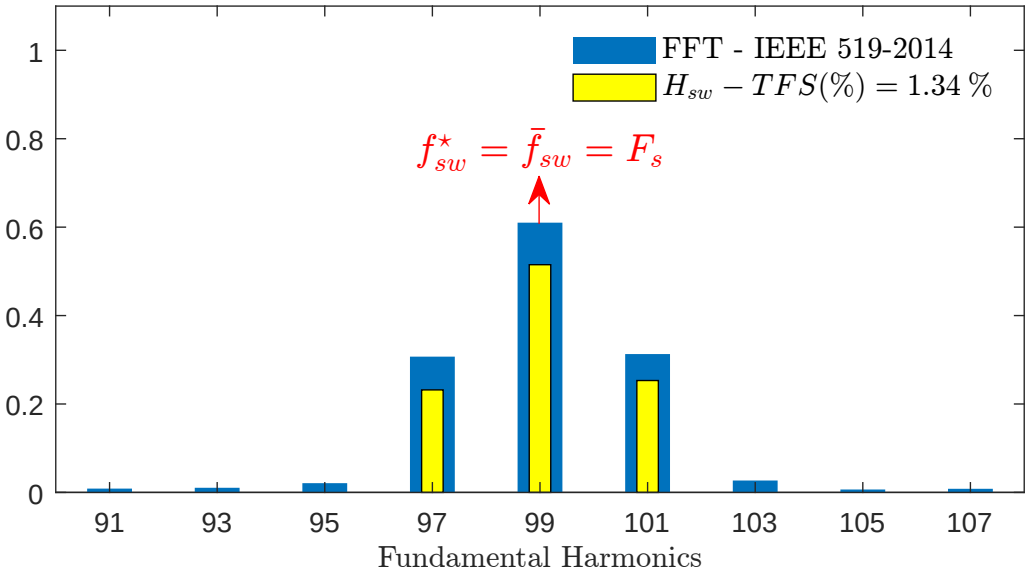


Figure 6.2: FFT of the switching pulses of a PWM-based converter and the instantaneous switching frequency components (H_{sw}).

6.1.3 Analysis of TFS in FCS-MPC

Figure 6.3 shows the estimation of the instantaneous switching frequency of an FCS-MPC with high variability: it ranges from a low switching frequency component (f_{sw}^{min}) to its maximum (f_{sw}^{max}) theoretical value, which equals half of the sampling frequency.

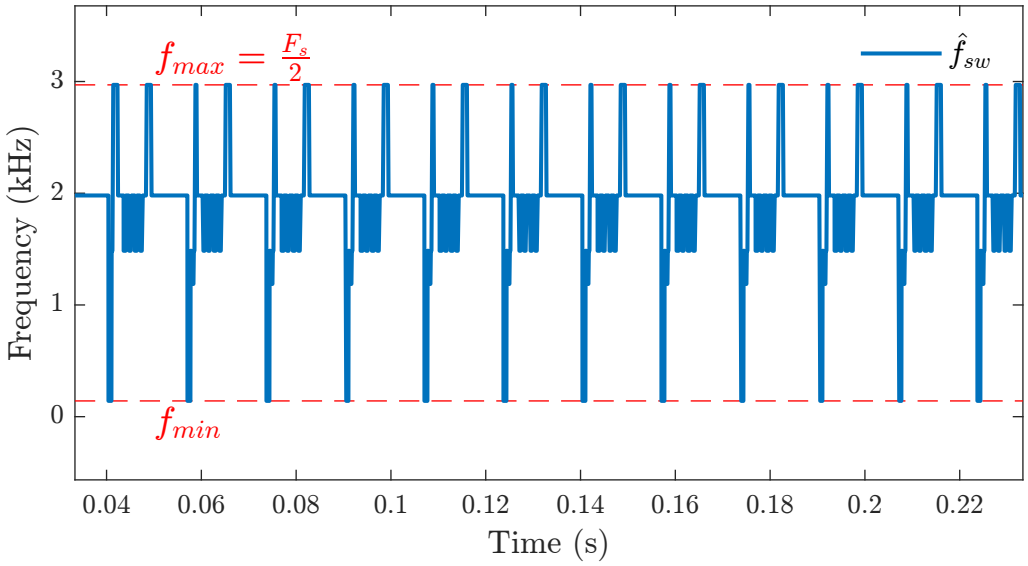


Figure 6.3: Instantaneous switching frequency of an FCS-MPC with sampling-frequency equal to 5940 Hz ($F_s = 99 \times 60\ Hz$).

The literature already established that the maximum switching frequency of FCS-MPC is half of the sampling frequency [1] — with a profile highly spread below this maximum value — but misses providing a quantitative method to analyze the spread frequencies.

Figure 6.4 (a) depicts the FFT of the switching pulses produced by FCS-MPC, while (b) shows the average switching frequency (yellow bar) and the set of switching frequency components (H_{sw} – red bars).

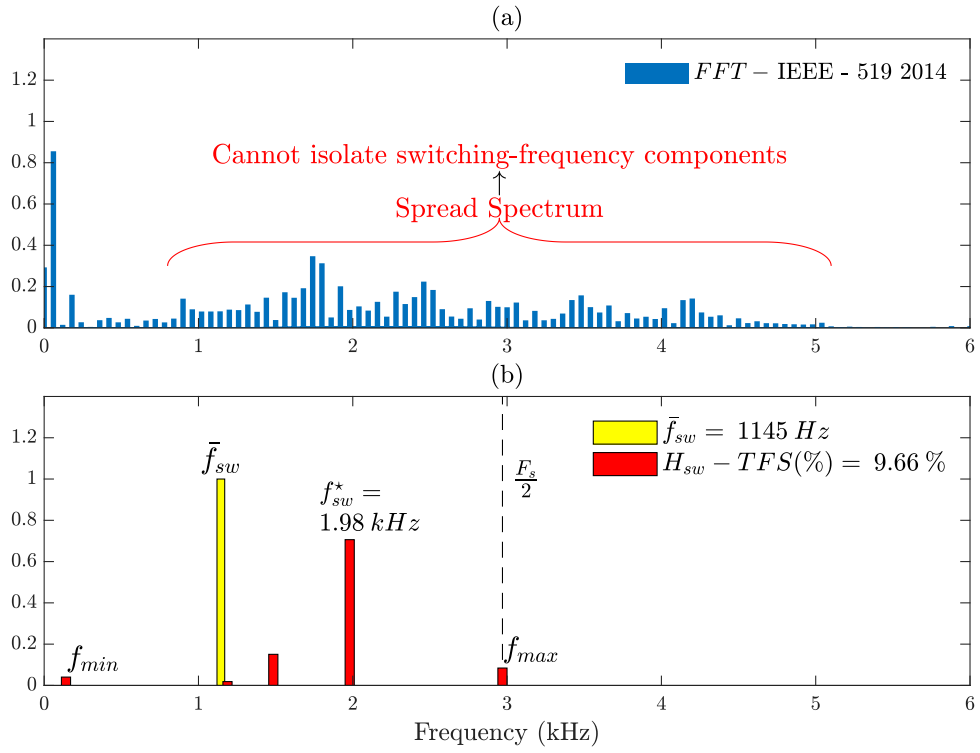


Figure 6.4: FCS-MPC with sampling frequency equal to 5940 Hz: (a) FFT of the switching pulses and (b) switching frequency components (H_{sw}).

One can note that a qualitative analysis of (a) makes it highly difficult to isolate the switching frequency components, unlike the case of a PWM-based converter.

However, the distribution of the FCS-MPC instantaneous switching frequency (see (b)) isolates the switching frequency components and shows that the average switching frequency, $\bar{f}_{sw}$, is not even a component of h_{sw}, and it is way distant from the dominant-frequency.

In other words, $\bar{f}_{sw}$ is only a poor average representation of the spread profile of switching pulses produced by FCS-MPC.

The spread profile of the switching frequency and the low-frequency component (f^{min}_{sw}) cause a typical issue in FCS-MPC: zero crossing error. Figure 6.5 displays a comparison between the waveforms of the converter ac-currents for both Jaya-MPC and FCS-MPC.

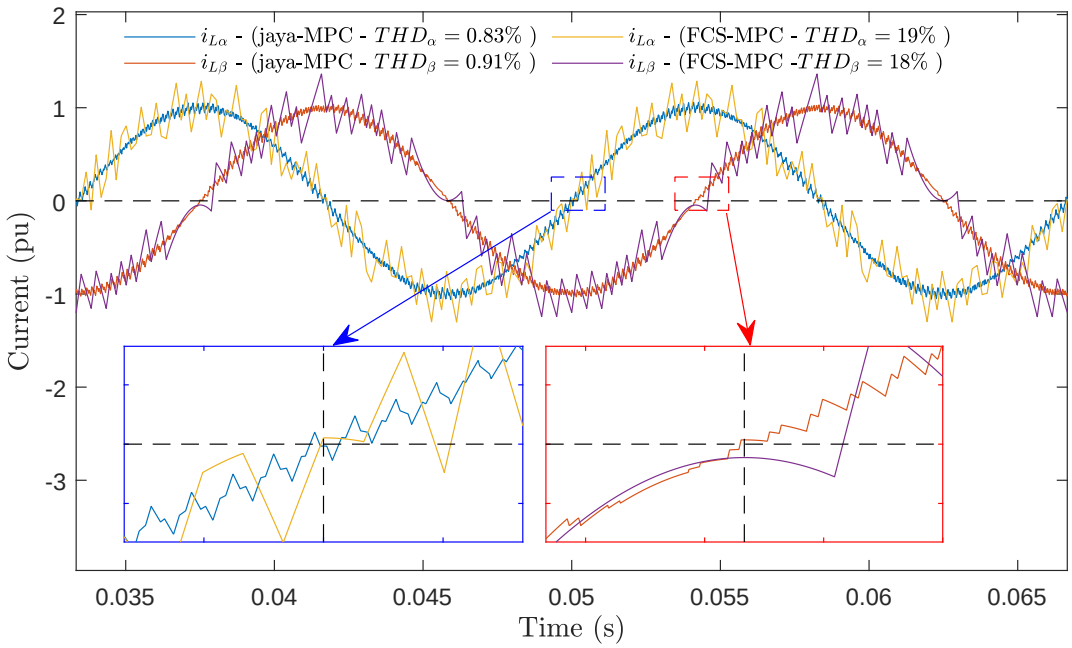


Figure 6.5: Comparison of the zero-crossing error between Jaya-MPC and FCS-MPC (with sampling frequency (F_s) of 5940 Hz).

The ripple in the ac-currents using the FCS-MPC is bigger and more irregular than the one of Jaya-MPC because all those spread components of Figure 6.4 (b); if the FCS-MPC had only the average switching frequency(1145 Hz), the ripples would not be so irregular.

These findings demonstrate the drawback of FCS-MPC, which is the variable switching frequency profile that makes it hard to design the filter, and may lead to undesired resonance effects if any of the switching frequency components are closer to the resonance frequency of the filter.

One could argue that increasing the sampling frequency would reduce the switching frequency spread in FCS-MPC since it decreases the THD. However, Figure 6.6 proves otherwise: it shows an overall trend of THD reduction as the sampling frequency increases, but the TFS presents a highly non-linear behavior, meaning that the spread profile of FCS-MPC highly depends on the sampling frequency choice, a result that the literature yet had not provided.

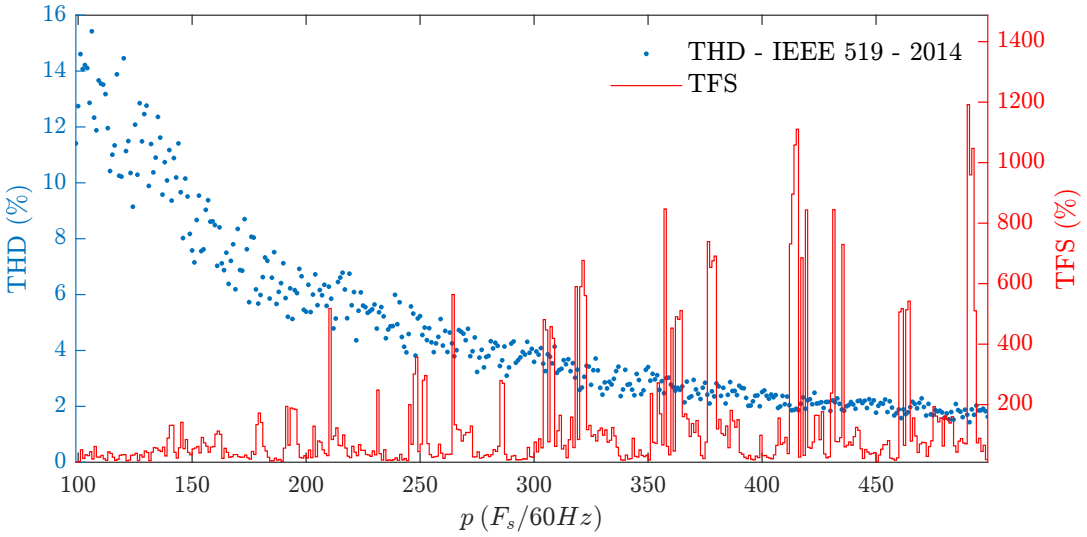


Figure 6.6: Sampling frequency effect in FCS-MPC: THD and TFS.

The analysis of the dominant frequency leads to the finding that this main component of the switching frequency profile is an integer fraction of the sampling frequency:

$$f_{sw}^{\star} = \frac{F_s}{q},\ \{q \in \mathbb{N}/q \geq 2\} \tag{6.6}$$

Figure 6.7 (a) confirms this finding by showing the value of $f_{sw}^{\star}$ (green dots and yellow circles) for the same set of sampling frequencies of the previous analysis — the dashed lines show the relations between F_s and $f_{sw}^{\star}$. In most cases, q ranges from 2 to 7 (see (b)), but for some sampling frequencies, the values of q are greater, making $f_{sw}^{\star}$ more distant from f_{sw}^{max}, therefore, increasing the TFS — these cases are the reason of those high TFS peaks of Figure 6.6. Once again, these results prove how difficult it is to set the sampling frequency of the FCS-MPC and how the spread of the switching frequency profile can be a consequence of this choice.

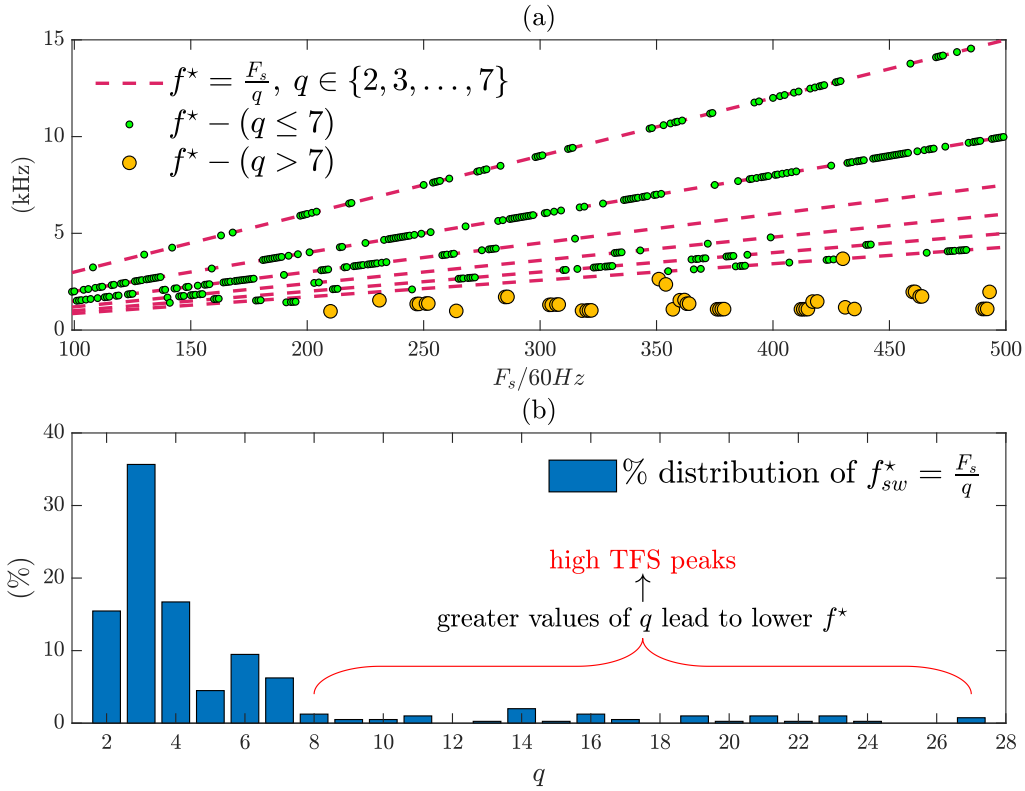


Figure 6.7: Effect of sampling frequency in the dominant frequency ($f^\star_{sw}$) of FCS-MPC: (a) $f^\star_{sw}$ as an integer ratio of F_s (see Equation (6.6)), for different values of q; (b) percent distribution of q — high TFS peaks are associated to high values of q.

Setting the sampling frequency as a multiple integer p of the fundamental frequency ($F_s = p \times 60Hz$) results in a dominant frequency proportional to the fundamental frequency:

$$f^\star_{sw} = \frac{p}{q} \times 60Hz,\ p >> q \tag{6.7}$$

If p and q were integer multiples of each other, it would ensure the dominant frequency is a harmonic component of the fundamental frequency. However, the previous results show that there is no easy way to determine q. Therefore, the dominant frequency often will be a non-characteristic component in the switching frequency profile.

All the previous works in the literature use either THD or the average switching frequency to characterize FCS-MPC, but now the findings of this thesis prove that these metrics are not suitable tools to analyze the switching frequency spread produced by FCS-MPC techniques.

Those works provide valuable contributions to the field but can only lead to qualitative observations, unlike this thesis, which presented the first metric to evaluate the switching frequency spread of FCS-MPC quantitatively.

6.2 Analysis On The Computational Cost

Although GPC techniques, such as Jaya-MPC, operate with a fixed switching frequency, their formulation and computational cost are more complex and costly. But, as Jaya-MPC stands out as a simple-formulation GPC technique, the last issue it must address is the computational cost.

6.2.1 Computational Cost Metric

This thesis uses a metric to compare the computational cost of MPC techniques. This metric quantifies how many predictions — and cost function evaluations — the MPC strategy assesses during a base period. Equation 6.8 shows this metric, called number-of-predictions-per-period (NPP), where n_p is the number of predictions that the controller assess during a normalized period (T_o), given a base period (T_{base}); the value of T_{base} is equal to one sampling-period of the PWM block (T_s) for the analyses of this section.

$$NPP = \frac{n_p}{T_o} \tag{6.8}$$

For the Jaya-MPC, the number of predictions during a sampling period (T_s) is equal to the number of generations (n_g) that it takes to return the optimal solution times the population size (M).

$$n_p = n_g \times \mathrm{M} \tag{6.9}$$

This work analyses both Jaya-MPC and FCS-MPC performing ac-current control in the $\alpha\beta$-frame. FCS-MPC evaluates the same number of predictions regardless of the axes, but Jaya-MPC spends different number of generations on each axis. Hence, it calculates different values of the number of predictions on each axis in the same sampling period. Therefore, Equation 6.9 counts the sum of the α- and β-axes predictions:

$$n_p = (n_{g\alpha} + n_{g\beta}) \times \mathrm{M} \tag{6.10}$$

For the Jaya-MPC, the normalized period is equal to one ($T_o = 1$) because the base period is equal to the sampling period ($T_s = T_{base}$). Hence, Equation 6.8 turns into the Equation 6.11:

$$NPP = (n_{g\alpha} + n_{g\beta}) \times \mathrm{M} \tag{6.11}$$

As the population size is constant, the value of NPP is a function of the number of generations. In this sense, the maximum value of NPP is:

$$max(NPP) = max\left(n_{g\alpha} + n_{g\beta}\right) \times \mathrm{M} \tag{6.12}$$

On the other hand, the number of predictions of FCS-MPC is proportional to the number of switching states of the converter (N_s). As the FCS-MPC controls the current of two axes ($\alpha\beta$-frame), Equation (6.13) computes the number of predictions for FCS-MPC:

$$n_p = 2N_s \tag{6.13}$$

Substituting (6.13) in (6.8), Equation (6.14) yields the value of NPP:

$$NPP = \frac{2N_s}{T_o} \tag{6.14}$$

Thus, one may assess the NPP for different values of sampling-frequency (f_k) in FCS-MPC through Equation (6.15), given the base period T_s.

$$NPP = 2N_s \times f_k \times T_s \tag{6.15}$$

6.2.2 Comparative Analyses for a Three-Phase Converter

Figure 6.8 (a) shows the number of generations that Jaya-MPC calculates for each one of the axes in the $\alpha\beta$-frame, while (b) presents the sum of these two quantities reflecting the total computational burden of the algorithm.

Based on Equations (6.11) and (6.15) it is possible to compare the computational cost of the Jaya-MPC and the FCS-MPC for a two-level three-phase converter — which has 8 switching states ($N_s = 2^3$) —, using different values of sampling-frequency in the FCS-MPC (see Table 6.1).

Table 6.1: Comparative analysis between Jaya-MPC and FCS-MPC regarding the computational cost.

	Jaya-MPC (F_s)	FCS-MPC ($f_k = k \times F_s$)			
		$k = 1$	$k = 2$	$k = 3$	$k = 4$
Sampling Frequency	5940 Hz	5940 Hz	11880 Hz	17820 Hz	23760 Hz
$max(NPP)$	36	16	32	48	64
THD	0.9%	18%	6.9%	3.7%	2.3%

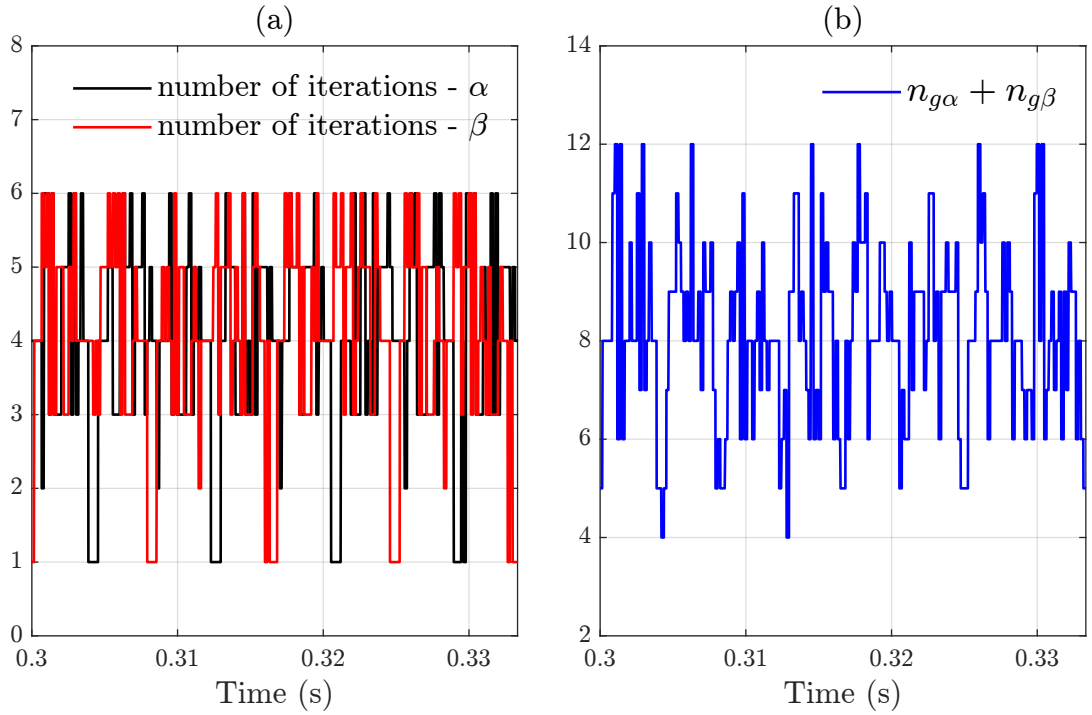


Figure 6.8: Number of generations of Jaya-MPC: (a) $n_{g\alpha}$ and $n_{g\beta}$, and (b) total number of generations ($n_{g\alpha} + n_{g\beta}$).

The results of this comparison lead to the following findings:

- the max(NPP) of Jaya-MPC is slightly superior to the one of FCS-MPC with a sampling frequency twice the Jaya-MPC switching frequency;
- But, when looking at THD, one can note that FCS-MPC needs greater sampling frequency to achieve THD lower than 5% — which is an expected behavior found in the literature;
- It means that Jaya-MPC demands a viable amount of predictions that can compete with FCS-MPC regarding the computational burden since FCS-MPC needs a sampling period greater than two times the switching frequency of Jaya-MPC, which is necessary to achieve lower THD.

These observations support the conclusion that besides inherently providing a fixed switching frequency profile with low THD, Jaya-MPC results in a solution with a computational cost, similar, or even better, to the one of the classical FCS-MPC technique that typically needs greater values of sampling frequency to achieve viable THD values —which reduces the available computational capability, i.e., the amount of processing time, to address all the predictions and cost function evaluations.

Chapter 7

Experimental Validation

Real-time validation of the proposed Jaya-MPC in a power converter is the main goal of this chapter. It describes the experimental setup, the development method, and the tests to validate the Jaya-MPC. First, it presents the equipment, software, and tools used in this stage of the work. Then, it explains the code development method that ensures code reliability. Further, the results, split into three sections, demonstrate the computational cost viability, current-regulation performance, and power quality, confirming that Jaya-MPC is a viable solution for real-time control of power converters.

7.1 Experimental Setup

The experimental Setup comprises a set of equipment, measurement instruments, and software available in the Laboratory of Power Electronics and Medium Voltage (LEMT).

Figure 7.1 shows the test bench and Table 7.1 describes each element in this setup. The circuit diagram of Figure 7.2 details how these elements interconnect. The dc-side power source is a variable transformer with a rectifier connected to the secondary winding. This equipment provides dc output voltage, manually regulated, connected to the dc-link of the power converter, which operates with a nominal power of 10 kW.

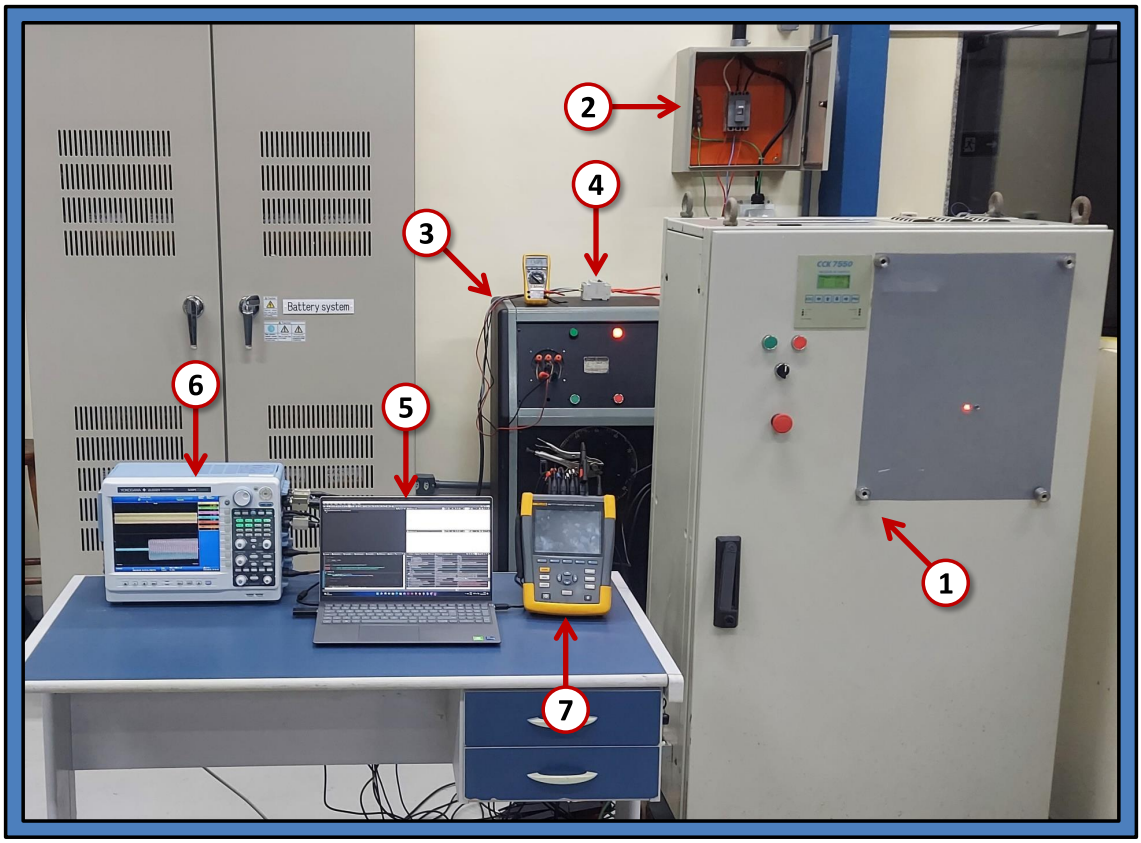


Figure 7.1: Equipment used in the experimental setup.

Two measurement instruments collected the data and waveforms during the experimental trials: an Oscilloscope, model DL850EV - YOKOGAWA, and a Power Quality and Energy Analyzer, model 435 Series II - FLUKE.

Code Compose Studio (CCS) is the Integrated Development Environment (IDE) used to develop and debug the embedded code programmed in a microcontroller (TMS320F28335 from *Texas Instruments*) that controls and automates the power converter.

Table 7.1: Experimental setup description.

	description
①	10 kW dc-ac IGBT-based two-level three-phase power converter
②	ac side circuit breaker — connects ① to the utility grid (220 V_{rms})
③	VARIAC plus rectifier — provides adjustable DC voltage to ①
④	circuit breaker — connects the ac side of ③ to the utility grid (440 V_{rms})
⑤	debugging session of CCS session connected with ① via USB cable
⑥	Oscilloscope – YOKOGAWA - DL850EV
⑦	Power Quality and Energy Analyzer – FLUKE - 435 Series II

The power converter has a set of 5 sensors (see Figure 7.2) to measure the ac line voltages (V_{ab}, V_{bc}), the ac currents (I_a,I_b), and the dc link voltage (V_{dc}); the inductor L; the ac side contactor K; and the control board , which reads the sensors' measurements and acts to turn-on/turn-off the contactor K.

Table 7.2 gives the nominal electrical parameters and the filter parameters for the experimental setup presented in Figure 7.2; it also provides the Jaya-MPC setup. Note that these are the same electrical system parameters of the simulations carried out in the previous chapters.

Table 7.2: Experimental setup parameters.

Nominal Electrical Parameters	
$V_{rms,LL}$	220 V
V_{dc}	450 V
S	10 KA
f	60 Hz
Filter Parameters	
L	2.03 mH
R	30.6 mΩ
Jaya-MPC Parameters	
N	8
tol	10^{-4}
Switching Frequency (F_{sw})	5940 Hz
Sampling Period ($\frac{1}{F_{sw}}$)	168.35 μs
weighting-factors strategy	correlated-adaptive
initial weight-factors	($r_1 = 0.33, r_2 = 0.33$)

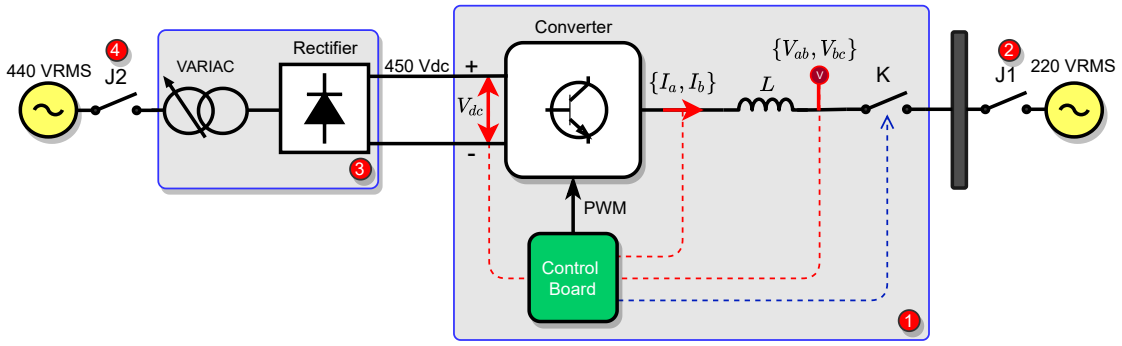


Figure 7.2: Circuit diagram of the experimental setup.

7.1.1 State Machine

The state machine of the CCS project is a C++ class that manages the automation of the converter and runs the Jaya-MPC controller. The flowchart of Figure 7.3 gives a simple understanding of the main code steps involved in the execution of this class:

- code starts in *Standby* state, which resets the relevant variables, ensuring the ac side contactor is open and the PWM module is off;
- if the user presses the start button, it transits to the *TurningOn* state that enables the PWM module and waits for 100 ms to transit to *Operating* state;
- by entering the *Operating* state, the code closes the ac side contactor, connecting the converter to the utility grid;
- while in *Operating* state, the embedded system executes the Jaya-MPC's C++ class, returning the modulation index to the PWM module;
- if any alarm triggers, the code switches to the *AlarmState*, which opens the ac side contactor and turns off the PWM module;
- if the cause of the alarm has ceased and the user has pressed the stop button, it switches back to *Standby*;
- if the user presses the stop button during regular operation, it also transits to *Standby*.

This state machine ensures the level of automation and protection that the experimental tests require.

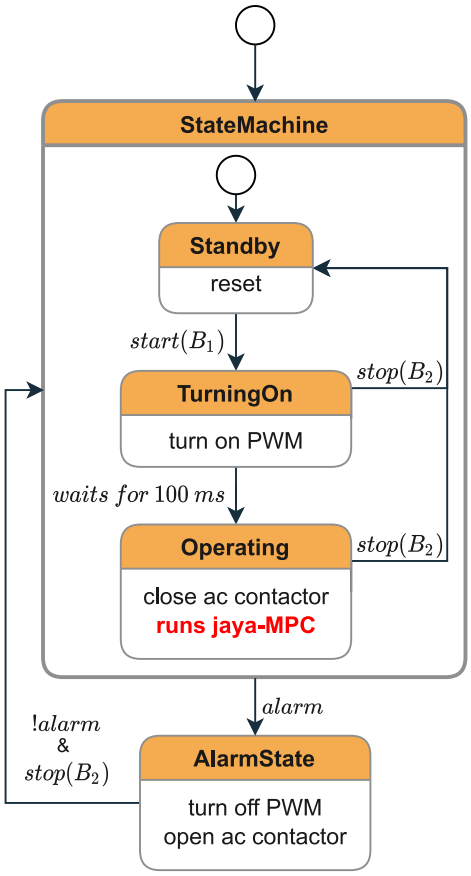


Figure 7.3: Flowchart of the state machine used in the CCS project.

7.2 Code Development in CCS

This section summarizes the method to develop the embedded code project, which comprises two main elements: the test-driven development (TDD) approach, including the description of the two major groups of tests, and the pipeline feature that joins these elements in an automated test platform.

7.2.1 Test Driven Development (TDD)

Test-Driven Development (TDD) is a simple yet powerful software development methodology that ensures reliability. Figure 7.4 illustrates this methodology: first, the programmer writes a small test that checks a feature he wants to add or change in a code; as this feature is yet missing, the test fails, and only after that the programmer can write a piece of code that makes the test pass. Then one can refactor the code to clean or organize it, but with the certainty that the test still passes after code changes; whenever the project demands a new feature, you shall create a new test.

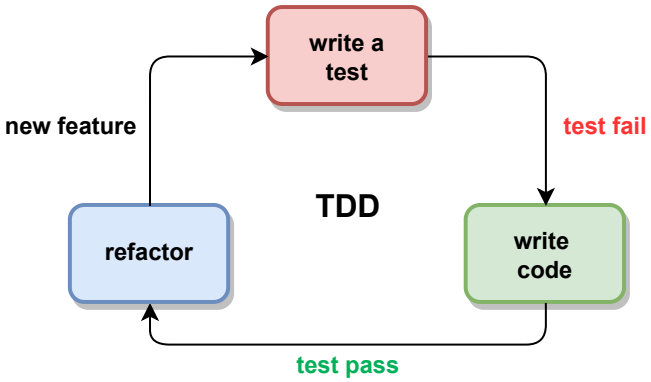


Figure 7.4: Simple representation of Test-Driven-Development (TDD) approach.

TDD aims to create a foundation in which it built the code project, ensuring most of the code has tests that verify its behavior. Therefore, if any future changes or updates occur, the tests secure the overall reliability of the code.

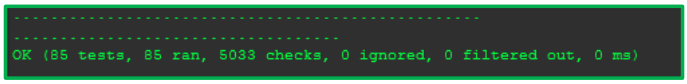
This work provides two main sets of tests to validate the overall performance and functionality of the embedded code: the unit tests, programmed in C++, check the features of each class in the CCS project. Figure 7.5 (a) shows the result of the log console of CCS after building the C++ project with all 85 test passing. The second group of tests comprises automatic $PSCAD^{TM}$ simulations that specify the overall control system performance and check if each alarm protects the converter within the specified parameters. Figure 7.5 (b) presents the result of these 20 tests passing.

Despite the TDD reliability characteristic, the overall development depends on an automatic trigger that can execute these batches of tests each time a modification occurs in code. Then, the method of this thesis uses a feature from the GitLab platform, called *Pipeline*, that triggers all these tests each time a commit occurs.

GitLab is an online platform that integrates several tools for code development, but the core of this platform is *version control* that allows the users to commit, i.e., to register all the changes made in the project, including the code, during the development steps.

Using the *pipeline* tool, whose interface is depicted in Figure 7.6, we can automatically execute all the C++ unit tests and $PSCAD^{TM}$ simulation tests; the pipeline tool builds the CCS project to check if any code change breaks the memory map of the microcontroller used in the power converter. This last test does not pass only if a code change or addition results in a compiled project that does not fit in the memory map of the microcontroller.

(a)



(b)

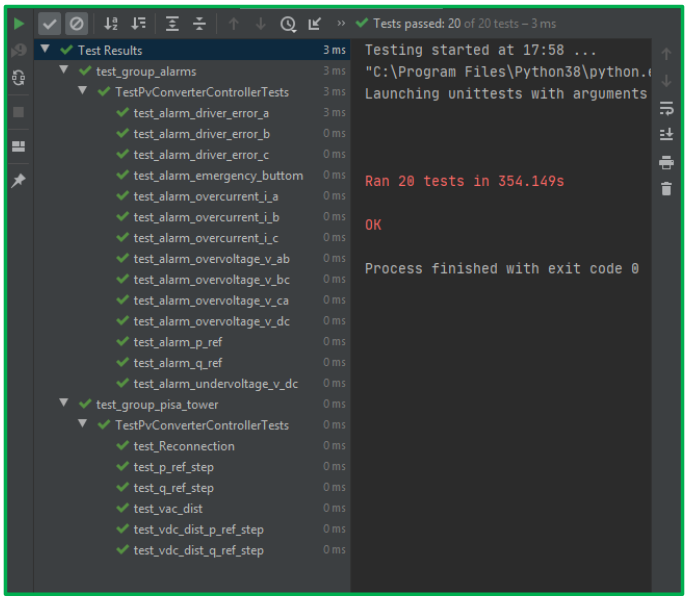


Figure 7.5: Log from (a) CCS after running C++ unit tests and (b) pyCharm after running $PSCAD^{TM}$ simulation tests.

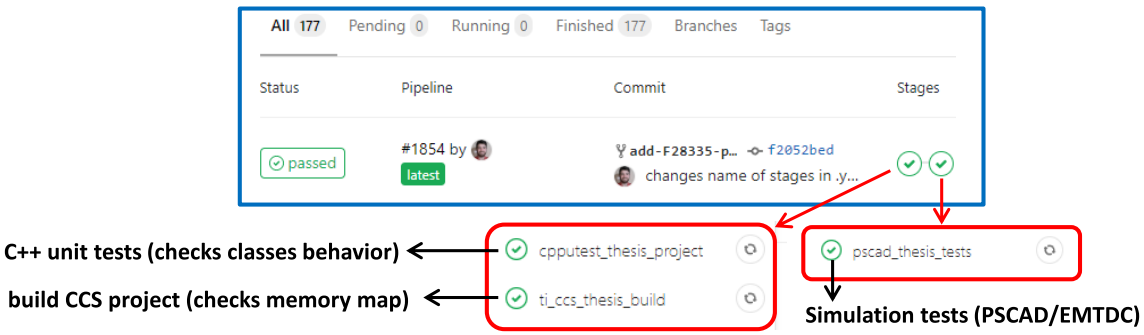


Figure 7.6: Pipeline tests ensuring expected code behavior.

7.3 Computational Cost

Figure 7.7 presents the computational cost and regulation error of the experimental results for different operation points. These data came from the graph functionality in Code Compose Studio (CCS) that plots data from the microcontroller while the embedded code is being executed in real-time. The horizontal axis of each plot contains a set of samples that CCS updates periodically through serial communication. It saves a set of 198 samples, which means two periods of the fundamental frequency with a sampling frequency of 5940 Hz.

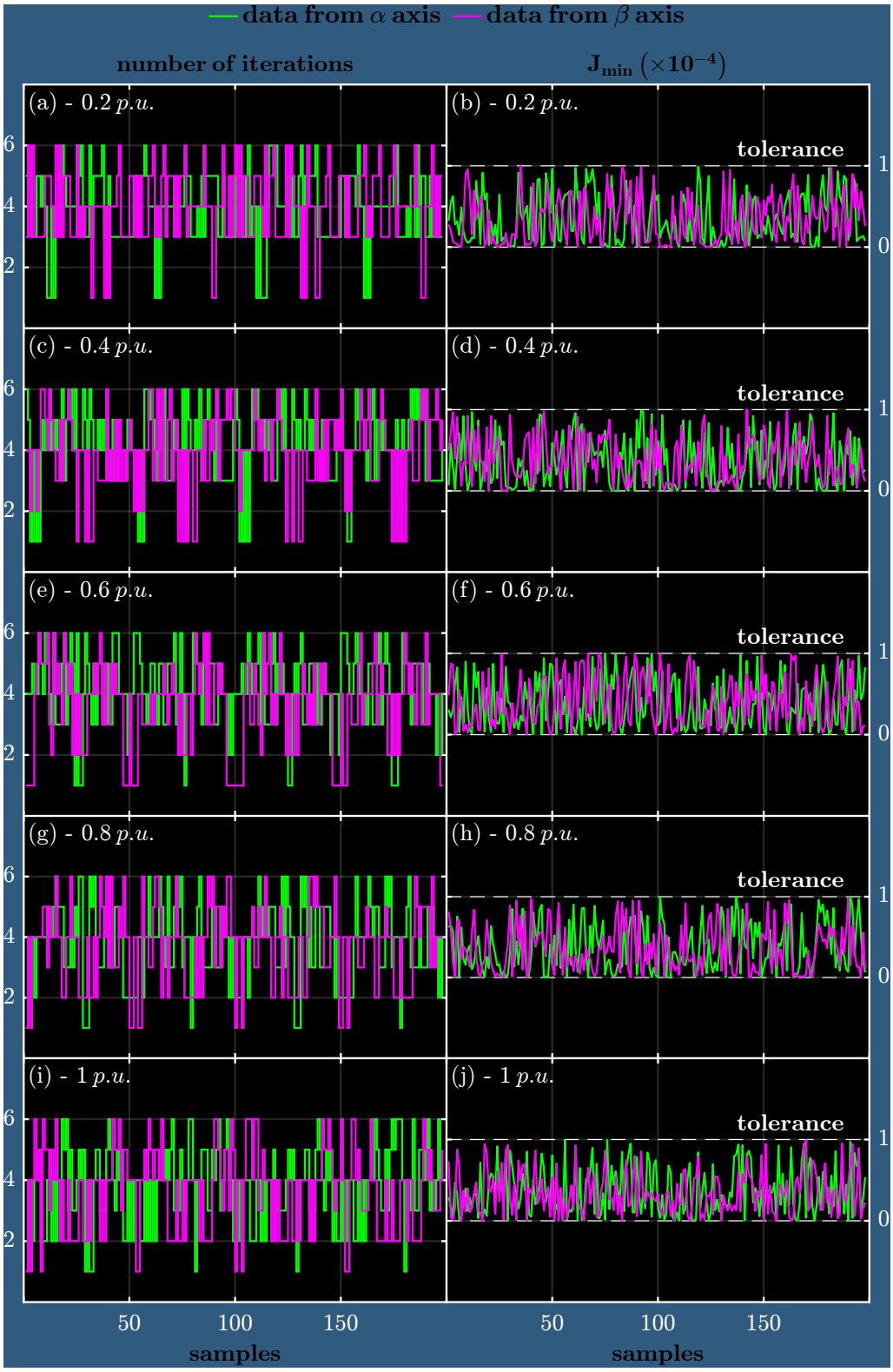


Figure 7.7: Experimental data from Code Compose Studio (CCS) showing the number of generations (left column) and minimum cost (right column) achieved by Jaya-MPC: for different ac current setpoints: (a) and (b) $i_{ref} = 0.2\,pu$; (c) and (d) $i_{ref} = 0.4\,pu$; (e) and (f) $i_{ref} = 0.6\,pu$; (g) and (h) $i_{ref} = 0.8\,pu$; (i) and (j) $i_{ref} = 1.0\,pu$.

The left column of Figure 7.7 shows the number of generations (left axis) per sampling period while the right column depicts the regulation error (right axis), i.e., the value of the minimum cost achieved by Jaya-MPC after executing the respective amount iterations each sampling period. These data confirm that the real-time operation of the proposed controller ensures a regulation error below the tolerance within a limited and well-defined number of generations.

Figure 7.8 shows the expected number of generations and regulation errors observed in the $PSCAD^{TM}$ — simulations run the same code project used in the experimental setup. Comparing this result against Figure 7.7, one can conclude that the simulation's methodology is robust and precise in replicating the behavior observed in the experimental setup.

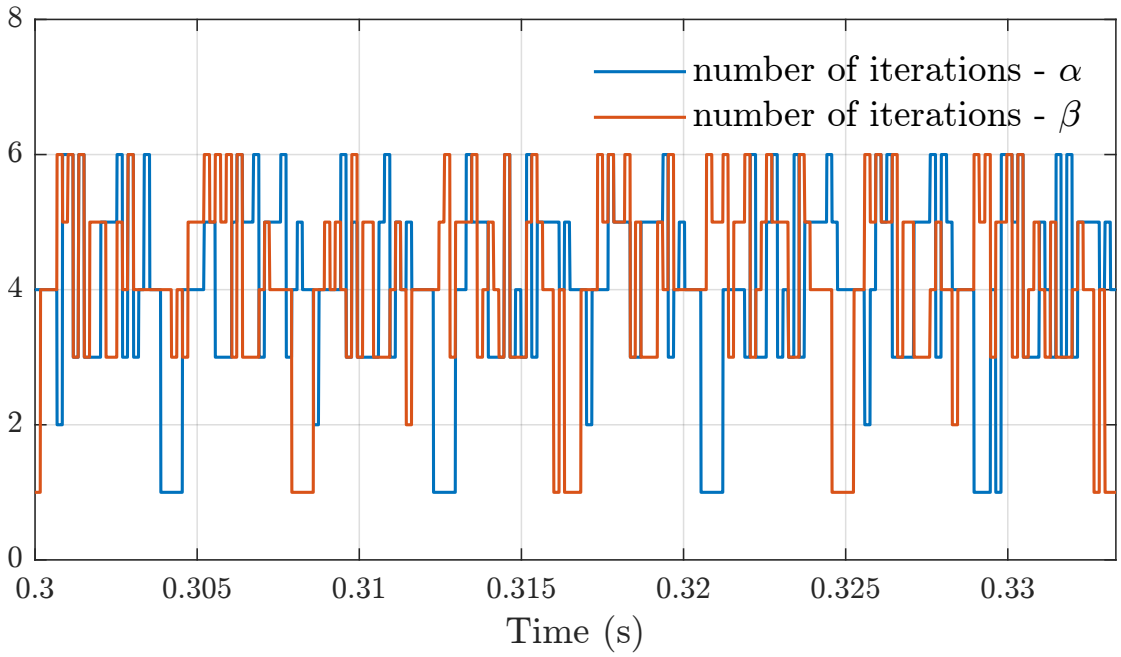


Figure 7.8: Simulation of the number of generations of Jaya-MPC in $PSCAD^{TM}$ with the same embedded code used in the experimental setup.

The outcomes presented in Figure 7.9 reinforce the real-time viability of the proposed control: in (a), it depicts the time that only the Jaya-MPC class takes to calculate the optimal control effort; in (b), it shows the time the entire embedded code project takes [11] to execute all the routines and return the modulation index.

In both cases, the execution time is variable since it depends on the number of generations the Jaya-MPC runs in each sampling period. But we can determine the maximum execution time using the oscilloscope's persistence configuration that overlaps each set of samples obtained when a trigger occurs. This acquisition mode

[11] Besides the control class, the embedded code includes the measurements class, the protection system, the state machine, the plot feature class that allows for plotting internal variables in the debug session of CCS, besides the references generation block.

guarantees the oscilloscope saves the waveforms during steady-state and transient, including the worst-case, overlapping them.

Using this acquisition mode, we can determine the total number of generations (see (c)) of Jaya-MPC — the sum of iterations from both α- and β- controllers — range from 4 up to 12 iterations. Hence, the maximum execution times (worst case) of the Jaya-MPC and the hole embedded code are 54 μs and 120 μs [12], respectively; both respect the sampling time of 168 μs and agree with data from Figure 7.7 and from the analysis of Section 6.2 (see Figure 6.8).

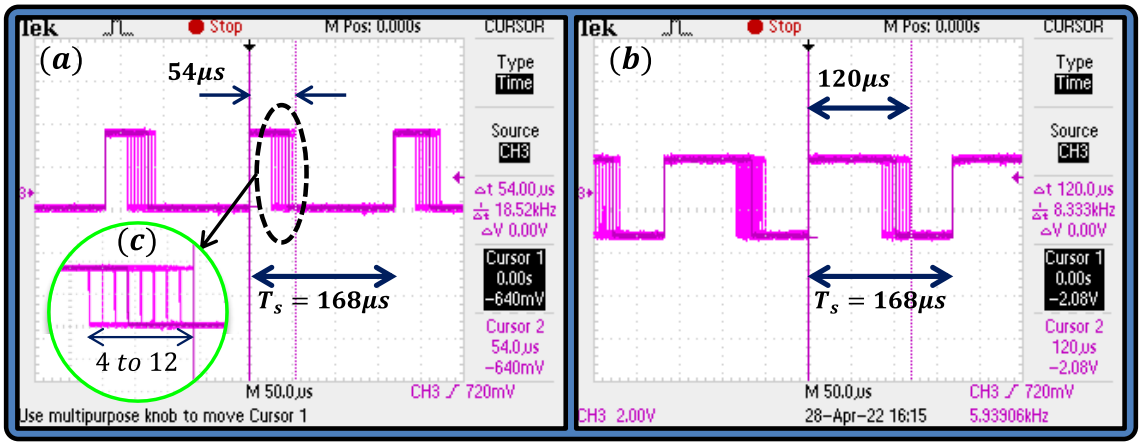


Figure 7.9: Execution times of (a) Jaya-MPC control class and (b) entire CCS project.

In summary, the experimental outcomes displayed in Figures 7.7 and 7.9 prove that the proposed Jaya-MPC ensures a regulation error below the desired tolerance within a well-determined number of generations, resulting in a viable computation cost approach, therefore, a viable control strategy for real-time application. Indeed, these processing times can be reduced by optimizing the code and compiled program.

[12]The embedded code was developed upon a C++ library that optimizes the code regarding memory usage. Therefore, the processing times were optimized — neither by the code structure nor by the compiler. It means that these processing times can be reduced; these results are a proof of concept that demonstrates the real-time operation viability of the Jaya-MPC algorithm.

7.4 Current Regulation Performance

The following outcomes, collected using a YOKOGAWA oscilloscope (model DL850EV), describe the initialization and turn-off processes of the converter and the ac-current regulation performance achieved by the Jaya-MPC.

Figure 7.10 (a) shows the instant time the PWM module activates, and after 100 ms, it closes the ac side contactor (see (b)), making the measured voltage equal to the grid voltage — this aim of this approach is to avoid current inrush when turning on the PWM module.

At the same time, the ac side contactor closes (see (b)), and the control receives a reference step of 1.0 pu of ac-current. Figure 7.10 (c) shows that the converter currents suffer no overshoot when the equipment is subject to a nominal active power step.

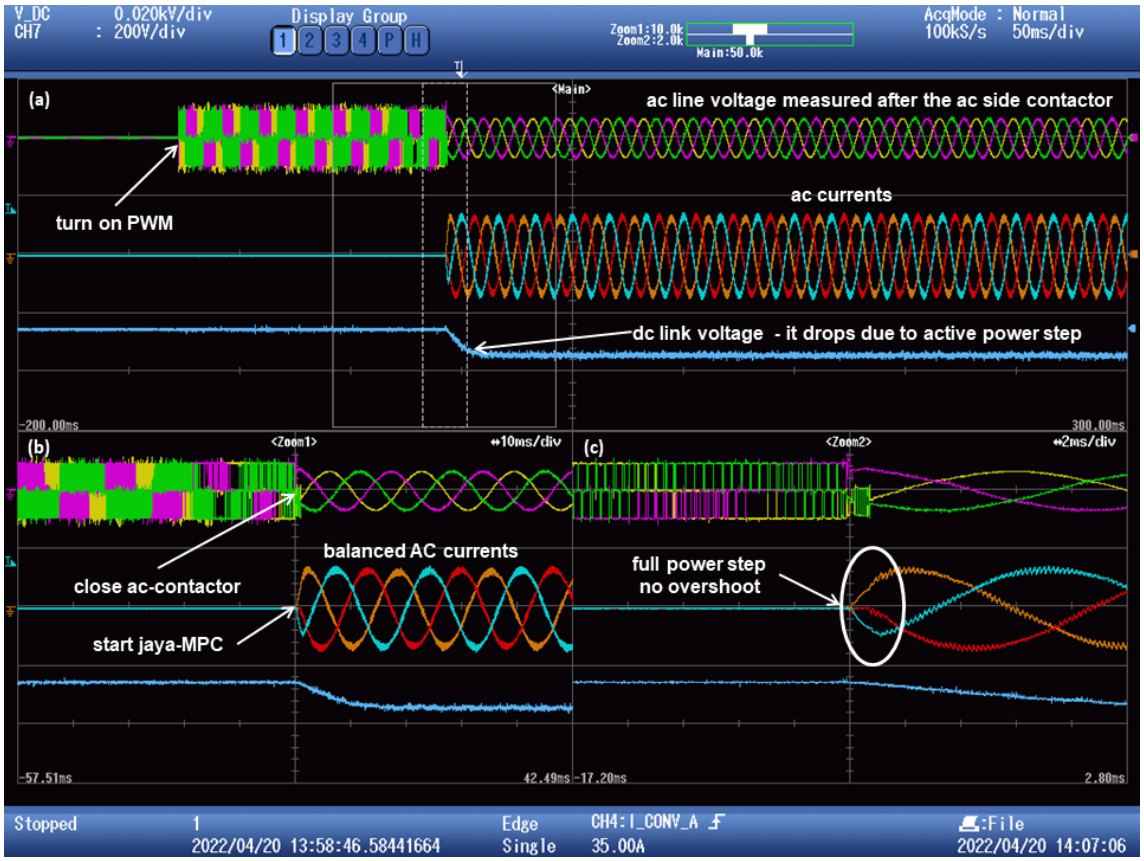


Figure 7.10: Experimental results: (a) turning on converter with active power reference of 1.0 pu; (b) Jaya-MPC starts after closing AC ; (c) zoom of the initial ac current transient with no overshoot.

The dc-link voltage drops when this power step occurs because the dc voltage source (VARIAC plus rectifier) has a considerable ac-side impedance that reduces the secondary winding voltage, causing a dc voltage disturbance. But, this disturbance does not affect the ac-current regulation during the transient period because MPC typically provides a high disturbance-rejection capability.

The simulation of the same scenario of Figure 7.10, depicted in Figure 7.11, confirms that the $PSCAD^{TM}$ simulation, using the same embedded code from the experimental setup, can replicate the results observed in Figure 7.10 (b), with almost identical transient response.

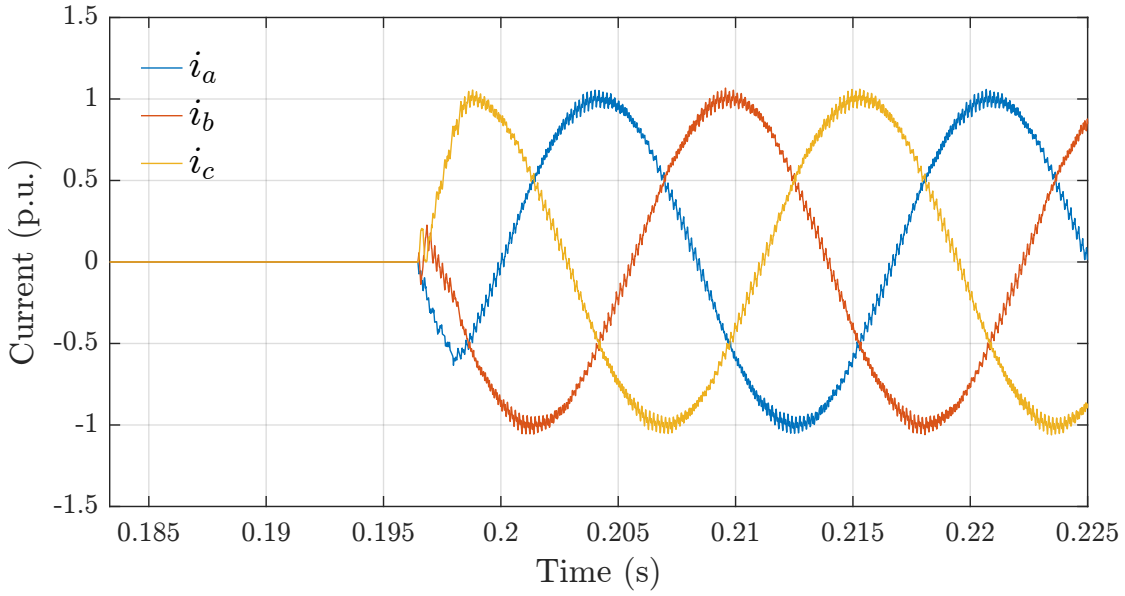


Figure 7.11: $PSCAD^{TM}$ Simulation of the converter ac currents in the same scenario of Figure 7.10, running the same embedded code used in the experimental setup.

Figure 7.12 shows a scenario in which a reactive current step occurs from $0.1\,pu$ to $1.0\,pu$. In(b), it shows the fast response of Jaya-MPC ac-current regulation, and, in (a), one can note that the dc-link voltage drop is almost null, different from the previous cases because now the system deals majority with reactive power, and only with lower active power, related to the converter losses.

The simulation result of Figure 7.13 shows this same scenario of reactive current injection. Again, The waveforms of Figure 7.12 (b) and Figure 7.13 present almost identical transient response.

Figure 7.12: Experimental results: (a) reactive power step from 0.1 pu to 1.0 pu; (b) zoom out of the transition period with no overshoot.

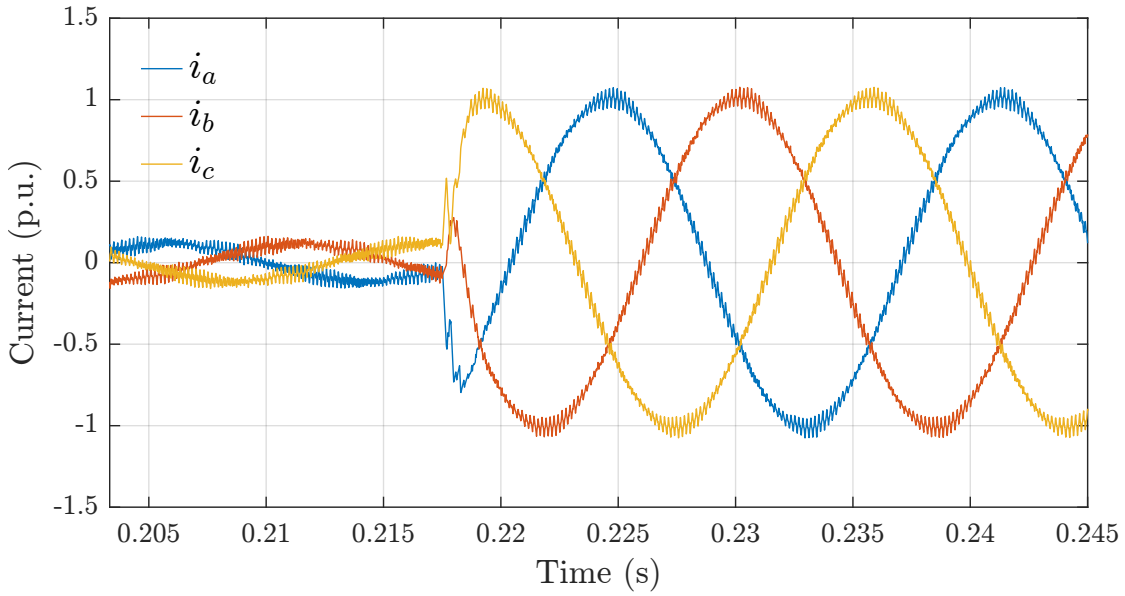


Figure 7.13: $PSCAD^{TM}$ Simulation of the converter ac currents in the same scenario of Figure 7.12, running the same embedded code used in the experimental setup.

Figure 7.14 presents a scenario where a 0.5 *pu* to 1.0 *pu* step followed by 1.0 *pu* to 0.5 *pu* down step occurs. The ac currents remain balanced during the test, and the dc-voltage disturbances do not affect the overall control performance.

Figure 7.14: Experimental results: (a) active power step from 0.5 *pu* to 1.0 *pu*, then back to 0.5 *pu*; (b) and (c) zoom out of the transition periods with no overshoot.

Figure 7.15 (a) shows the smooth turn-off dynamics of the converter, disabling the PWM module and disconnecting the equipment from the ac grid. Here, the ac currents go from nominal active power to zero with no issue (see (b)); the dc-voltage disturbance is because of the same reason explained in the previous cases and, again, it causes no safety problem for the equipment.

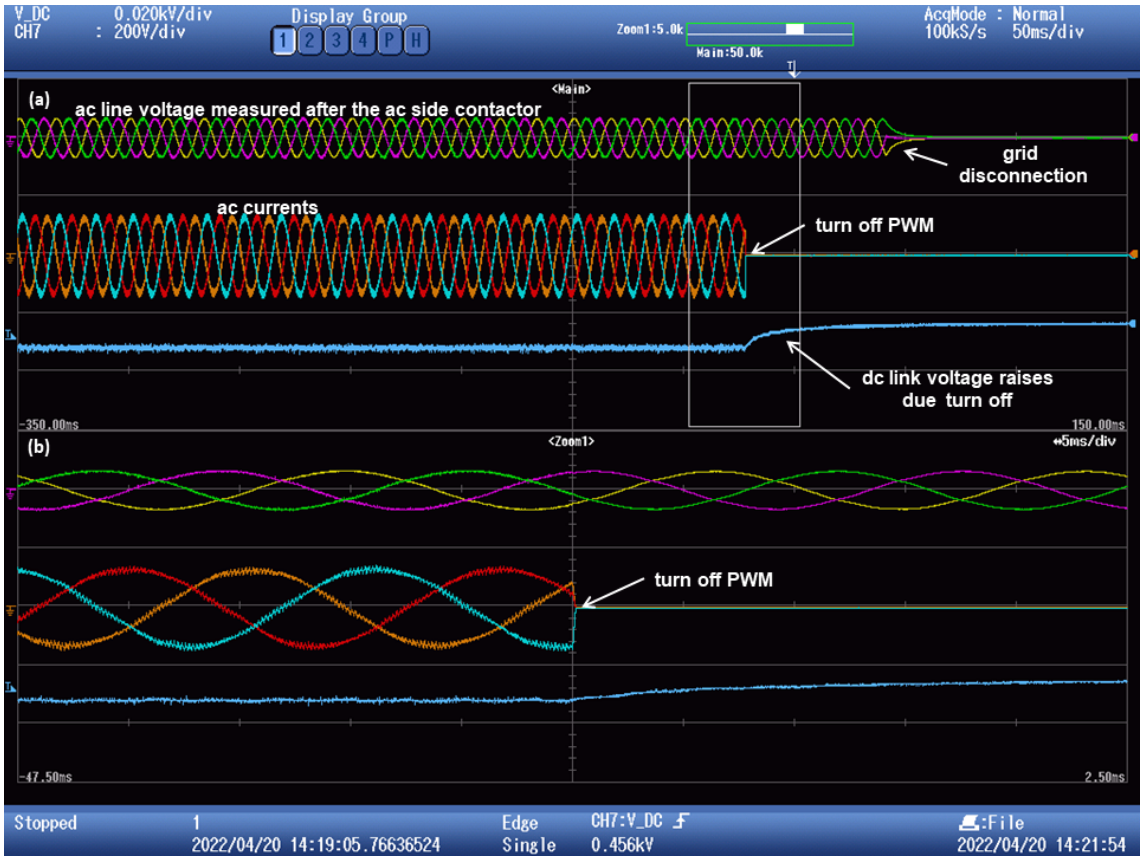


Figure 7.15: Experimental results: (a) turning off converter with an active power step from 1.0 pu to 0.0 pu; (b) zoom of instant time that code turns off PWM.

7.5 Power Quality

This section presents the experimental results, collected using a Power Quality and Energy Analyzer (model FLUKE - 435 Series), regarding power quality achieved by Jaya-MPC.

The experimental setup is subjected to a grid line voltage profile of Figure 7.16 (a) with low harmonic content (see (b)). This stage of the experimental validation comprises two tests, one for active-power injection and the other for reactive-power injection; both analyze the THD and the harmonic content of the converter ac currents in steady-state.

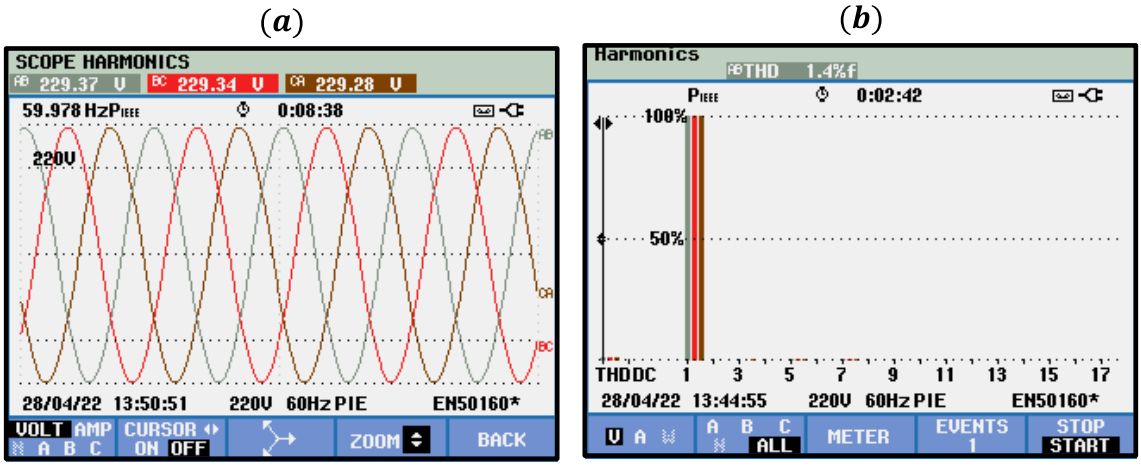


Figure 7.16: Experimental data collected using FLUKE - 435 Series II: (a) grid line voltage waveforms and (b) their harmonic content.

Figure 7.17 (a) and (b) show the phasor-planes of the test cases in which the converter injects nominal active power and reactive power, respectively — the same scenarios of the steady-state stage of Figures 7.10 and 7.12. The thick arrows represent the voltage phasors, while the thin arrows represent the current phasors; (c) and (d) display the harmonic content of the ac currents from both cases. These outcomes confirm that Jaya-MPC produces a high power-quality solution, but active power injection results in lower THD (2.2% – see (c)) than the one of reactive power injection (3.3% – see (d)).

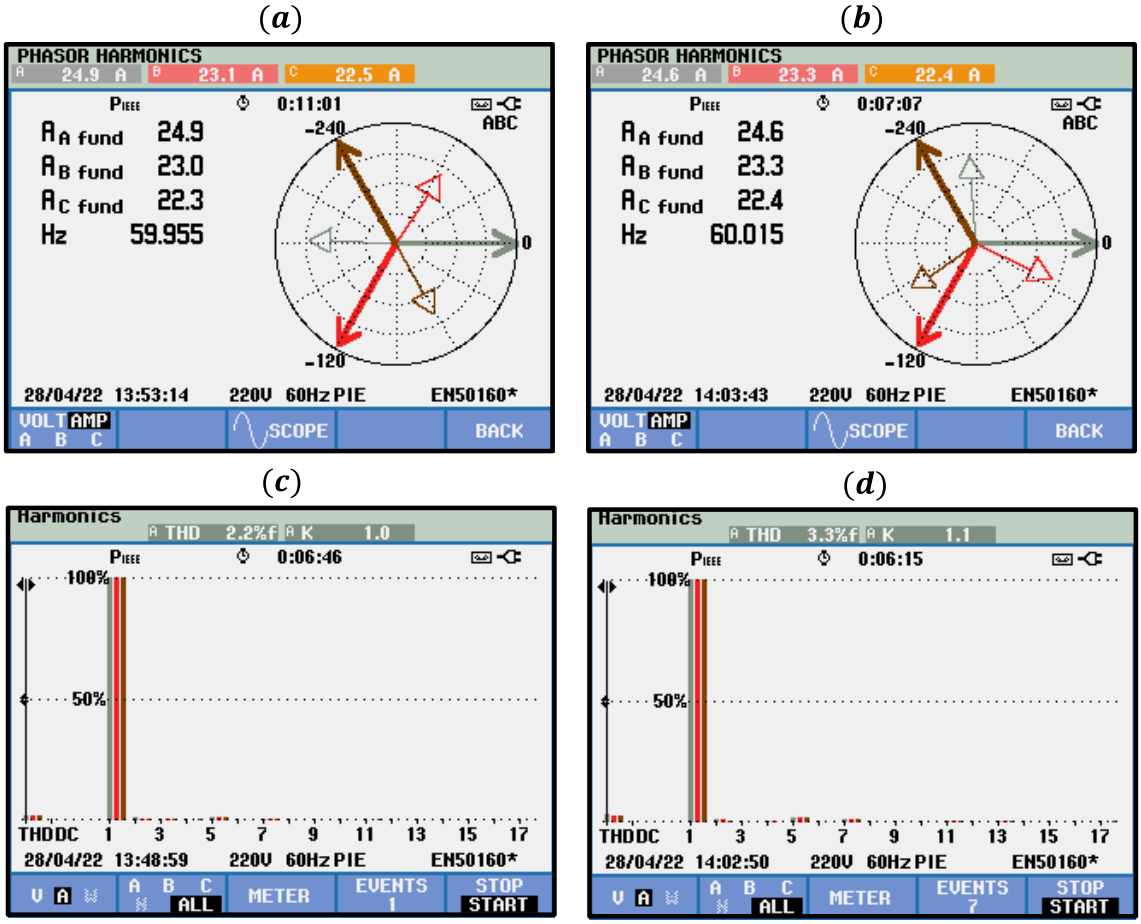


Figure 7.17: Experimental data collected using FLUKE - 435 Series II: ac current phasors for (a) active power injection and (b) reactive power injection, and their respective harmonic contents (c) and (d).

Chapter 8

Conclusions and Future Works

8.1 Conclusion

Keeping researching alternatives to improve the control performance of power converters is a major demand in the power electronics field. MPC is a path for these improvements and research. This work followed this path and, based on the literature review on MPC in power electronics, formulated the following question:

Is it possible to design a GPC strategy proposing an optimization solver that makes its formulation and implementation simple and reduces the computational cost to values close to the ones observed in FCS-MPC?

This question motivated the research of this thesis and, according to the literature review presented in Chapter 3, the Jaya algorithm stood out as a promising solution since it does not need any algorithm-specific hyper-parameters, dismisses the use of derivatives in the formulation, and solves a large variety of optimization problems.

Thus, this thesis proposed an MPC strategy that uses a modified Jaya algorithm as the controller optimization solver. The proposed algorithm aims to be a viable alternative to FCS-MPC regarding implementation complexity and reliable computational cost. It is worth pointing out that the findings of this work are restricted to the scope of the main application of MPC in power electronics, which is ac-current control of grid-connected power converters.

Although the original Jaya algorithm presents no algorithm-specific hyper-parameters, the proposed Jaya-MPC demands the setting of one weight factor because of the changes to remove the non-deterministic behavior from the original algorithm.

The parametric analyses described in Chapter 5 lead to the finding that the adaptive-weight approach outperforms the fixed-weight one, providing a simple design solution. This approach ensures low THD in the ac currents and a viable

number of generations. The consequence of this last feature is that Jaya-MPC's computational cost can compete with the one of FCS-MPC since the latter often demands high sampling frequencies to achieve better regulation performance.

This work also introduced a new metric (TFS) to analyze the spread of the switching frequency profile of FCS-MPC. The proposed metric suits this analysis better because it is based on an instantaneous estimation of the switching frequency. Future researchers can now use these TFS-based analyses to assess the switching frequency spread of MPC-based strategies.

The results of Chapter 6 supported the conclusion that FCS-MPC works with a high TFS that is very sensitive to the choice of the sampling period and that the average switching frequency proved to be a poor tool for specifying FCS-MPC. This work also showed that the main switching frequency component often is a non-characteristic harmonic of the fundamental frequency.

These results reinforce the argument that designing viable and simple-implementation GPC strategies is a possible alternative to FCS-MPC; This thesis did not analyze other FCS-MPC-derived approaches, like those that fall into the OSS-MPC classification. But, as these strategies are more complex versions of implementation and formulation than FCS-MPC and have higher computational costs than FCS-MPC when Jaya-MPC competes with FCS-MPC in the matter of computational cost, it shows to be a viable alternative for predictive control applications in power electronics converters within the scope of this thesis proposal. It is important to note that the processing time of a control strategy is highly dependent on the hardware and the code development approach, as well as the optimization of the code. This work shows that Jaya-MPC operates in real-time, proving to be a viable alternative to the field.

The experimental results and the methodology of this research, including the automatic test system approach, prove that the Jaya-MPC works with high power-quality performance, dismissing the use of extra control loops to reject harmonics and an affordable real-time computational cost.

Therefore, by combining the outcomes and findings from these analyses, this work concludes that Jaya-MPC is a viable and competitive alternative to FCS-MPC, presenting high-performance dynamics with fixed-switching frequency and low computational cost.

8.2 Future Works

The continuation of this work includes extending the proposed Jaya-MPC to other applications in power electronics, such as LCL-filter-based converters and LC-filter-based converters operating as voltage sources, i.e., with voltage control. The proposed controller presents a valuable potential for Multilevel Converters. Code development of the proposed technique can include optimization practices to reduce the processing time of the proposed controller.

Concerning the evaluation of the switching frequency spread, future studies can analyze the TFS of other MPC strategies to advance the findings regarding this subject in power electronics applications. Furthermore, TFS can be quantified along with the switching losses of MPC-based power converters to determine a relation between the spread and the converter's efficiency.

References

[1] VAZQUEZ, S., FRANQUELO, L. G., NORAMBUENA, M., et al. "Model Predictive Control for Power Converters and Drives: Advances and Trends", *IEEE Transactions on Industrial Electronics*, v. 64, n. 2, pp. 935–947, 2016. ISSN: 0278-0046. doi: 10.1109/tie.2016.2625238.

[2] KOURO, S., PEREZ, M. A., RODRIGUEZ, J., et al. "Model Predictive Control: MPC's Role in the Evolution of Power Electronics", *IEEE Industrial Electronics Magazine*, v. 9, n. 4, pp. 8–21, 12 2015. ISSN: 1932-4529.

[3] KARAMANAKOS, P., GEYER, T. "Guidelines for the Design of Finite Control Set Model Predictive Controllers", *IEEE Transactions on Power Electronics*, v. 35, n. 7, pp. 7434–7450, 7 2020. ISSN: 0885-8993.

[4] KARAMANAKOS, P., LIEGMANN, E., GEYER, T., et al. "Model Predictive Control of Power Electronic Systems: Methods, Results, and Challenges", *IEEE Open Journal of Industry Applications*, v. 1, n. June, pp. 95–114, 2020. ISSN: 2644-1241.

[5] RODRIGUEZ, J., CORTES, P., KENNEL, R., et al. "Model predictive control – a simple and powerful method to control power converters". In: *2009 IEEE 6th International Power Electronics and Motion Control Conference*, v. 56, pp. 41–49. IEEE, 5 2009. ISBN: 978-1-4244-3556-2.

[6] RODRIGUEZ, J., KAZMIERKOWSKI, M. P., ESPINOZA, J. R., et al. "State of the Art of Finite Control Set Model Predictive Control in Power Electronics", *IEEE Transactions on Industrial Informatics*, v. 9, n. 2, pp. 1003–1016, 5 2013. ISSN: 1551-3203.

[7] GAMOUDI, R., ELHAK CHARIAG, D., SBITA, L. "A Review of Spread-Spectrum-Based PWM Techniques - A Novel Fast Digital Implementation", *IEEE Transactions on Power Electronics*, v. 33, n. 12, pp. 10292–10307, dec 2018. ISSN: 0885-8993.

[8] JOSE RODRIGUEZ, P. C. *Predictive Control of Power Converters and Electrical Drives*. Wiley, 2012. ISBN: 9781119963981.

[9] VARGAS, R., CORTES, P., AMMANN, U., et al. "Predictive Control of a Three-Phase Neutral-Point-Clamped Inverter", *IEEE Transactions on Industrial Electronics*, v. 54, n. 5, pp. 2697–2705, 10 2007. ISSN: 0278-0046.

[10] CORTES, P., RODRIGUEZ, J., QUEVEDO, D. E., et al. "Predictive Current Control Strategy With Imposed Load Current Spectrum", *IEEE Transactions on Power Electronics*, v. 23, n. 2, pp. 612–618, 3 2008. ISSN: 0885-8993.

[11] AGGRAWAL, H., LEON, J. I., FRANQUELO, L. G., et al. "Model predictive control based selective harmonic mitigation technique for multilevel cascaded H-bridge converters", *IECON Proceedings (Industrial Electronics Conference)*, pp. 4427–4432, 2011.

[12] STELLATO, B., GEYER, T., GOULART, P. J. "High-Speed Finite Control Set Model Predictive Control for Power Electronics", *IEEE Transactions on Power Electronics*, v. 32, n. 5, pp. 4007–4020, 5 2017. ISSN: 0885-8993.

[13] VAZQUEZ, S., MARQUEZ, A., AGUILERA, R., et al. "Predictive Optimal Switching Sequence Direct Power Control for Grid-Connected Power Converters", *IEEE Transactions on Industrial Electronics*, v. 62, n. 4, pp. 2010–2020, 4 2015. ISSN: 0278-0046.

[14] VAZQUEZ, S., AGUILERA, R. P., ACUNA, P., et al. "Model Predictive Control for Single-Phase NPC Converters Based on Optimal Switching Sequences", *IEEE Transactions on Industrial Electronics*, v. 63, n. 12, pp. 7533–7541, 12 2016. ISSN: 0278-0046.

[15] TARISCIOTTI, L., ZANCHETTA, P., WATSON, A., et al. "A new predictive control method for cascaded multilevel converters with intrinsic modulation scheme", *IECON Proceedings (Industrial Electronics Conference)*, pp. 5764–5769, 2013.

[16] DONOSO, F., MORA, A., CARDENAS, R., et al. "Finite-Set Model-Predictive Control Strategies for a 3L-NPC Inverter Operating With Fixed Switching Frequency", *IEEE Transactions on Industrial Electronics*, v. 65, n. 5, pp. 3954–3965, 5 2018. ISSN: 0278-0046.

[17] AGUIRRE, M., KOURO, S., ROJAS, C. A., et al. "Switching Frequency Regulation for FCS-MPC Based on a Period Control Approach", *IEEE Transactions on Industrial Electronics*, v. 65, n. 7, pp. 5764–5773, 7 2018. ISSN: 0278-0046.

[18] TRICARICO, T., COSTA, J. A., HERRERA, D., et al. "Total Frequency Spread: A New Metric to Assess the Switching Frequency Spread of FCS-MPC", *Energies*, v. 15, n. 14, pp. 5273, jul 2022. ISSN: 1996-1073. doi: 10.3390/en15145273. Disponível em: <https://www.mdpi.com/1996-1073/15/14/5273>.

[19] RODRIGUEZ, J., KAZMIERKOWSKI, M. P., ESPINOZA, J. R., et al. "State of the Art of Finite Control Set Model Predictive Control in Power Electronics", *IEEE Transactions on Industrial Informatics*, v. 9, n. 2, pp. 1003–1016, may 2013. ISSN: 1551-3203. doi: 10.1109/TII.2012.2221469.

[20] KOURO, S., PEREZ, M. A., RODRIGUEZ, J., et al. "Model Predictive Control: MPC's Role in the Evolution of Power Electronics", *IEEE Industrial Electronics Magazine*, v. 9, n. 4, pp. 8–21, 2015. ISSN: 19324529. doi: 10.1109/MIE.2015.2478920.

[21] HOLTZ, J. "A predictive controller for the stator current vector of AC machines fed from a switched voltage source", *Proc. of IEE of Japan IPEC-Tokyo'83*, pp. 1665–1675, 1983.

[22] KENNEL, R., SCHRÖDER, D. "Predictive Control Strategy for Converters", *IFAC Proceedings Volumes*, v. 16, n. 16, pp. 415–422, 1983. ISSN: 14746670. doi: 10.1016/s1474-6670(17)61897-4.

[23] HOANG LE-HUY, SLIMANI, K., VIAROUGE, P. "Analysis and implementation of a real-time predictive current controller for permanent-magnet synchronous servo drives", *IEEE Transactions on Industrial Electronics*, v. 41, n. 1, pp. 110–117, 1994. ISSN: 02780046. doi: 10.1109/41.281616.

[24] RODRÍGUEZ, J., PONTT, J., CORREA, P., et al. "Predictive power control of an AC/DC/AC converter", *Conference Record - IAS Annual Meeting (IEEE Industry Applications Society)*, v. 2, n. 1, pp. 934–939, 2005. ISSN: 01972618. doi: 10.1109/IAS.2005.1518458.

[25] CORTES, P., RODRIGUEZ, J., VARGAS, R., et al. "Cost Function-Based Predictive Control for Power Converters", *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, pp. 2268–2273, nov 2006. ISSN: 1553-572X. doi: 10.1109/IECON.2006.347996.

[26] CORTÉS, P., RODRÍGUEZ, J. "Three-phase inverter with output LC filter using predictive control for UPS applications", *2007 European Conference*

on Power Electronics and Applications, EPE, 2007. doi: 10.1109/EPE.2007.4417385.

[27] CORTÉS, P., RODRÍGUEZ, J., QUEVEDO, D., et al. "Predictive current control strategy with imposed load current spectrum", *EPE-PEMC 2006: 12th International Power Electronics and Motion Control Conference, Proceedings*, v. 23, n. 2, pp. 252–257, 2007. doi: 10.1109/EPEPEMC.2006.283088.

[28] RIVERA, M. E., VARGAS, R. E., ESPINOZA, J. R., et al. "Behavior of the predictive DTC based matrix converter under unbalanced AC supply", *Conference Record - IAS Annual Meeting (IEEE Industry Applications Society)*, pp. 202–207, 2007. ISSN: 01972618. doi: 10.1109/07ias.2007.16.

[29] RODRIGUEZ, J., PONTT, J., SILVA, C. A., et al. "Predictive Current Control of a Voltage Source Inverter", *IEEE Transactions on Industrial Electronics*, v. 54, n. 1, pp. 495–503, feb 2007. ISSN: 0278-0046. doi: 10.1109/TIE.2006.888802.

[30] VARGAS, R., CORTES, P., AMMANN, U., et al. "Predictive Control of a Three-Phase Neutral-Point-Clamped Inverter", *IEEE Transactions on Industrial Electronics*, v. 54, n. 5, pp. 2697–2705, oct 2007. ISSN: 0278-0046. doi: 10.1109/TIE.2007.899854.

[31] CORREA, P., RODRIGUEZ, J. "A predictive control scheme for current source rectifiers", *2008 13th International Power Electronics and Motion Control Conference, EPE-PEMC 2008*, pp. 699–702, 2008. doi: 10.1109/EPEPEMC.2008.4635346.

[32] VARGAS, R., AMMANN, U., RODRÍGUEZ, J., et al. "Predictive strategy to reduce common-mode voltages on power converters". In: *PESC Record - IEEE Annual Power Electronics Specialists Conference*, pp. 3401–3406, 2008. ISBN: 9781424416684. doi: 10.1109/PESC.2008.4592481.

[33] CORTÉS, P., KOURO, S., ROCCA, B. L., et al. "Guidelines for weighting factors design in model predictive control of power converters and drives", *Proceedings of the IEEE International Conference on Industrial Technology*, 2009. doi: 10.1109/ICIT.2009.4939742.

[34] KOURO, S., CORTES, P., VARGAS, R., et al. "Model Predictive Control A Simple and Powerful Method to Control Power Converters", *Industrial Electronics, IEEE Transactions on*, v. 56, n. 6, pp. 1826–1838, 2009. ISSN: 0278-0046. doi: 10.1109/TIE.2008.2008349.

[35] KENNEL, R., LINDER, A. "Predictive control of inverter supplied electrical drives", *2000 IEEE 31st Annual Power Electronics Specialists Conference. Conference Proceedings*, v. 2, n. 4, pp. 761–766, 2000. ISSN: 1098-6596. doi: 10.1109/PESC.2000.879911.

[36] KENNEL, R., LINDER, A., LINKE, M. "Generalized Predictive Control (GPC) - Ready for use in drive applications?" *PESC Record - IEEE Annual Power Electronics Specialists Conference*, v. 4, pp. 1839–1844, 2001. ISSN: 02759306. doi: 10.1109/pesc.2001.954389.

[37] LINDER, A., KENNEL, R. "Model predictive control for electrical drives", *PESC Record - IEEE Annual Power Electronics Specialists Conference*, v. 2005, pp. 1793–1799, 2005. ISSN: 02759306. doi: 10.1109/PESC.2005.1581874.

[38] LINDER, A., KENNEL, R. "Direct model predictive control - A new direct predictive control strategy for electrical drives", *2005 European Conference on Power Electronics and Applications*, v. 2005, 2005. doi: 10.1109/epe.2005.219335.

[39] RODRÍGUEZ, J., PONTT, J., SILVA, C., et al. "Predictive current control of a voltage source inverter", *PESC Record - IEEE Annual Power Electronics Specialists Conference*, v. 3, pp. 2192–2196, 2004. ISSN: 02759306. doi: 10.1109/PESC.2004.1355460.

[40] CATUCCI, M., CLARE, J., WHEELER, P. "Predictive control strategies for ZCS direct converter HV power supply", *2005 European Conference on Power Electronics and Applications*, v. 2005, pp. 1–10, 2005. doi: 10.1109/epe.2005.219519.

[41] PERANTZAKIS, G. S., XEPAPAS, F. H., PAPATHANASSIOU, S. A., et al. "A predictive current control technique for three-level NPC voltage source inverters", *PESC Record - IEEE Annual Power Electronics Specialists Conference*, v. 2005, pp. 1241–1246, 2005. ISSN: 02759306. doi: 10.1109/PESC.2005.1581788.

[42] PERANTZAKIS, G. S., XEPAPAS, F. H., MANIAS, S. N. "Efficient predictive current control technique for multilevel voltage source inverters", *2005 European Conference on Power Electronics and Applications*, v. 2005, n. it i, pp. 1–10, 2005. doi: 10.1109/epe.2005.219543.

[43] CATUCCI, M., CLARE, J., WHEELER, P. "Predective Control Strategy for ZCS Single Stage Resonant Converter". In: *IECON 2006 - 32nd Annual*

Conference on IEEE Industrial Electronics, v. 16, pp. 2905–2910. IEEE, nov 2006. ISBN: 1-4244-0390-1. doi: 10.1109/IECON.2006.347414.

[44] SILVA, E. I., MCGRATH, B. P., QUEVEDO, D. E., et al. "Predictive Control of a Flying Capacitor Converter". In: *2007 American Control Conference*, pp. 3763–3768. IEEE, jul 2007. ISBN: 1-4244-0988-8. doi: 10.1109/ACC.2007.4282860.

[45] GATTO, G., MARONGIU, I., SERPI, A., et al. "A predictive direct torque control of induction machines", *SPEEDAM 2008 - International Symposium on Power Electronics, Electrical Drives, Automation and Motion*, pp. 1103–1108, 2008. doi: 10.1109/SPEEDHAM.2008.4581129.

[46] VARGAS, R., RIVERA, M., RODRIGUEZ, J., et al. "Predictive torque control with input PF correction applied to an induction machine fed by a matrix converter". In: *2008 IEEE Power Electronics Specialists Conference*, pp. 9–14. IEEE, jun 2008. ISBN: 978-1-4244-1667-7. doi: 10.1109/PESC.2008.4591888.

[47] VARGAS, R., AMMANN, U., RODRIGUEZ, J., et al. "Reduction of switching losses and increase in efficiency of power converters using predictive control". In: *2008 IEEE Power Electronics Specialists Conference*, pp. 1062–1068. IEEE, jun 2008. ISBN: 978-1-4244-1667-7. doi: 10.1109/PESC.2008.4592070.

[48] GONTIJO, G. F., TRICARICO, T. C., FRANCA, B. W., et al. "Robust Model Predictive Rotor Current Control of a DFIG Connected to a Distorted and Unbalanced Grid Driven by a Direct Matrix Converter", *IEEE Transactions on Sustainable Energy*, v. 10, n. 3, pp. 1380–1392, 2019. ISSN: 19493029. doi: 10.1109/TSTE.2018.2868406.

[49] TRICARICO, T. C. *ESTUDO DE TÉCNICAS DE CONTROLE PARA INTERFACE DE POTÊNCIA DE UMA MICRORREDE HÍBRIDA UTILIZANDO UM CONVERSOR INTERLEAVED*. Tese de Mestrado, Universidade Federal do Rio de Janeiro, may 2018.

[50] YOUNG, H. A., PEREZ, M. A., RODRIGUEZ, J., et al. "Assessing finite-control-set model predictive control: A comparison with a linear current controller in two-level voltage source inverters", *IEEE Industrial Electronics Magazine*, v. 8, n. 1, pp. 44–52, 2014. ISSN: 19324529. doi: 10.1109/MIE.2013.2294870.

[51] RODRIGUEZ, J., CORTES, P. *Predictive Control of Power Converters and Electrical Drives*. Chichester, UK, John Wiley & Sons, Ltd, 2012. ISBN: 9781119941446. doi: 10.1002/9781119941446.

[52] YARAMASU, V., YU, B. *Model Predictive Control of Wind Energy Conversion Systems*. John Wiley & Sons, Ltd, 2017. ISBN: 978-1-118-98858-9.

[53] GEYER, T. *Model predictive control of high power converters and industrial drives*. John Wiley & Sons, Ltd, 2016. ISBN: 978-1-119-01090-6.

[54] KHALILZADEH, M., VAEZ-ZADEH, S., RODRIGUEZ, J., et al. "Model-Free Predictive Control of Motor Drives and Power Converters: A Review", *IEEE Access*, v. 9, pp. 105733–105747, 2021. ISSN: 21693536. doi: 10.1109/ACCESS.2021.3098946.

[55] BORDONS, C., MONTERO, C. "Basic Principles of MPC for Power Converters: Bridging the Gap Between Theory and Practice", *IEEE Industrial Electronics Magazine*, v. 9, n. 3, pp. 31–43, sep 2015. ISSN: 1932-4529. doi: 10.1109/MIE.2014.2356600.

[56] VAZQUEZ, S., AGUILERA, R. P., ACUNA, P., et al. "Model Predictive Control for Single-Phase NPC Converters Based on Optimal Switching Sequences", *IEEE Transactions on Industrial Electronics*, v. 63, n. 12, pp. 7533–7541, dec 2016. ISSN: 0278-0046. doi: 10.1109/TIE.2016.2594227.

[57] ZHENG, C., DRAGICEVIC, T., ZHANG, Z., et al. "Model Predictive Control of LC -Filtered Voltage Source Inverters With Optimal Switching Sequence", *IEEE Transactions on Power Electronics*, v. 36, n. 3, pp. 3422–3436, mar 2021. ISSN: 0885-8993. doi: 10.1109/TPEL.2020.3015540.

[58] RIVERA, M., RODRIGUEZ, J., WU, B., et al. "Current control for an indirect matrix converter with filter resonance mitigation", *IEEE Transactions on Industrial Electronics*, v. 59, n. 1, pp. 71–79, 2012. ISSN: 02780046. doi: 10.1109/TIE.2011.2165311.

[59] GEYER, T., KARAMANAKOS, P., KENNEL, R. "On the benefit of long-horizon direct model predictive control for drives with LC filters", *2014 IEEE Energy Conversion Congress and Exposition, ECCE 2014*, pp. 3520–3527, 2014. doi: 10.1109/ECCE.2014.6953879.

[60] PANTEN, N., HOFFMANN, N., FUCHS, F. W. "Finite Control Set Model Predictive Current Control for Grid-Connected Voltage-Source Converters

with LCL Filters: A Study Based on Different State Feedbacks", *IEEE Transactions on Power Electronics*, v. 31, n. 7, pp. 5189–5200, 2016. ISSN: 08858993. doi: 10.1109/TPEL.2015.2478862.

[61] AGUILERA, R. P., QUEVEDO, D. E. "Stability analysis of quadratic MPC with a discrete input alphabet", *IEEE Transactions on Automatic Control*, v. 58, n. 12, pp. 3190–3196, 2013. ISSN: 00189286. doi: 10.1109/TAC.2013.2264551.

[62] GEYER, T., AGUILERA, R. P., QUEVEDO, D. E. "On the stability and robustness of model predictive direct current control", *Proceedings of the IEEE International Conference on Industrial Technology*, pp. 374–379, 2013. doi: 10.1109/ICIT.2013.6505701.

[63] AGUILERA, R. P., QUEVEDO, D. E. "Predictive control of power converters: Designs with guaranteed performance", *IEEE Transactions on Industrial Informatics*, v. 11, n. 1, pp. 53–63, 2015. ISSN: 15513203. doi: 10.1109/TII.2014.2363933.

[64] VAZQUEZ, S., MONTERO, C., BORDONS, C., et al. "Design and experimental validation of a Model Predictive Control strategy for a VSI with long prediction horizon", *IECON Proceedings (Industrial Electronics Conference)*, pp. 5788–5793, 2013. doi: 10.1109/IECON.2013.6700083.

[65] KIEFERNDORF, F., KARAMANAKOS, P., BADER, P., et al. "Model predictive control of the internal voltages of a five-level active neutral point clamped converter", *2012 IEEE Energy Conversion Congress and Exposition, ECCE 2012*, v. 4, pp. 1676–1683, 2012. doi: 10.1109/ECCE.2012.6342611.

[66] MOON, J. W., GWON, J. S., PARK, J. W., et al. "Model Predictive Control with a Reduced Number of Considered States in a Modular Multilevel Converter for HVDC System", *IEEE Transactions on Power Delivery*, v. 30, n. 2, pp. 608–617, 2015. ISSN: 08858977. doi: 10.1109/TPWRD.2014.2303172.

[67] NORAMBUENA, M., DIECKERHOFF, S., KOURO, S., et al. "Finite control set model predictive control of a stacked multicell converter with reduced computational cost", *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, pp. 1819–1824, 2015. doi: 10.1109/IECON.2015.7392365.

[68] VATANI, M., BAHRANI, B., SAEEDIFARD, M., et al. "Indirect Finite Control Set Model Predictive Control of Modular Multilevel Converters", *IEEE Transactions on Smart Grid*, v. 6, n. 3, pp. 1520–1529, may 2015. ISSN: 1949-3053. doi: 10.1109/TSG.2014.2377112.

[69] YOUNG, H. A., PEREZ, M. A., RODRIGUEZ, J., et al. "Assessing Finite-Control-Set Model Predictive Control: A Comparison with a Linear Current Controller in Two-Level Voltage Source Inverters", *IEEE Industrial Electronics Magazine*, v. 8, n. 1, pp. 44–52, 3 2014. ISSN: 1932-4529.

[70] RAMIREZ, R. O., ESPINOZA, J. R., VILLARROEL, F., et al. "A Novel Hybrid Finite Control Set Model Predictive Control Scheme With Reduced Switching", *IEEE Transactions on Industrial Electronics*, v. 61, n. 11, pp. 5912–5920, 11 2014. ISSN: 0278-0046.

[71] TARISCIOTTI, L., ZANCHETTA, P., WATSON, A., et al. "Modulated Model Predictive Control for a Seven-Level Cascaded H-Bridge Back-to-Back Converter", *IEEE Transactions on Industrial Electronics*, v. 61, n. 10, pp. 5375–5383, 10 2014. ISSN: 0278-0046.

[72] ALHASHEEM, M., BLAABJERG, F., DAVARI, P. "Performance assessment of grid forming converters using different finite control set model predictive control (FCS-MPC) algorithms". 2019. ISSN: 20763417.

[73] ALHASHEEM, M., ABDELHAKIM, A., BLAABJERG, F., et al. "Model predictive control of grid forming converters with enhanced power quality", *Applied Sciences (Switzerland)*, v. 10, n. 18, 2020. ISSN: 20763417.

[74] VENKATA RAO, R. "Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems", *International Journal of Industrial Engineering Computations*, v. 7, n. 1, pp. 19–34, 2016. ISSN: 19232934. doi: 10.5267/j.ijiec.2015.8.004.

[75] PANDEY, H. M. "Jaya a novel optimization algorithm: What, how and why?" *Proceedings of the 2016 6th International Conference - Cloud System and Big Data Engineering, Confluence 2016*, pp. 728–730, 2016. doi: 10.1109/CONFLUENCE.2016.7508215.

[76] RAO, R. V., RAI, D. P. "Optimization of submerged arc welding process parameters using quasi-oppositional based Jaya algorithm", *Journal of Mechanical Science and Technology*, v. 31, n. 5, pp. 2513–2522, 2017. ISSN: 1738494X. doi: 10.1007/s12206-017-0449-x.

[77] RAO, R. V., MORE, K. C. "Design optimization and analysis of selected thermal devices using self-adaptive Jaya algorithm", *Energy Conversion and Management*, v. 140, pp. 24–35, 2017. ISSN: 01968904. doi: 10.1016/j.enconman.2017.02.068.

[78] RAO, R. V., RAI, D. P., BALIC, J. "A multi-objective algorithm for optimization of modern machining processes", *Engineering Applications of Artificial Intelligence*, v. 61, n. March, pp. 103–125, 2017. ISSN: 09521976. doi: 10.1016/j.engappai.2017.03.001.

[79] RAO, R. V., SAROJ, A. "An elitism-based self-adaptive multi-population Jaya algorithm and its applications", *Soft Computing*, v. 23, n. 12, pp. 4383–4406, 2019. ISSN: 14337479. doi: 10.1007/s00500-018-3095-z.

[80] VENKATA RAO, R., SAROJ, A. "A self-adaptive multi-population based Jaya algorithm for engineering optimization", *Swarm and Evolutionary Computation*, v. 37, n. October 2016, pp. 1–26, 2017. ISSN: 22106502. doi: 10.1016/j.swevo.2017.04.008.

[81] RAO, R. V. *Jaya: An advanced optimization algorithm and its engineering applications*. Cham, Springer International Publishing, 2018. ISBN: 9783319789224. doi: 10.1007/978-3-319-78922-4.

[82] AZIZIPANAH-ABARGHOOEE, R., MALEKPOUR, M., ZARE, M., et al. "A new inertia emulator and fuzzy-based LFC to support inertial and governor responses using Jaya algorithm", *IEEE Power and Energy Society General Meeting*, v. 2016-Novem, pp. 1–5, 2016. ISSN: 19449933. doi: 10.1109/PESGM.2016.7741906.

[83] MANDAL, S., MANDAL, K. K., KUMAR, S. "A new optimization technique for optimal reactive power scheduling using Jaya algorithm", *2017 Innovations in Power and Advanced Computing Technologies, i-PACT 2017*, v. 2017-Janua, pp. 1–5, 2017. doi: 10.1109/IPACT.2017.8244961.

[84] ACHANTA, R. K., PAMULAVINAY, K. "DC Motor Speed Control using PID Controller Tuned by Jaya Optimization Algorithm University College of Engineering Kakinada", *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, pp. 983–987, 2017.

[85] AKBARIMAJD, A., ASEFI, S., SHAYEGHI, H. "Using Jaya algorithm to optimal tuning of LQR based power system stabilizers", *2017 2nd IEEE International Conference on Computational Intelligence and Applications,*

ICCIA 2017, v. 2017-Janua, n. ii, pp. 482–486, 2017. doi: 10.1109/CIAPP.2017.8167264.

[86] HUANG, C., ZHANG, Z., WANG, L., et al. "A novel global maximum power point tracking method for PV system using Jaya algorithm", *2017 IEEE Conference on Energy Internet and Energy System Integration, EI2 2017 - Proceedings*, v. 2018-Janua, pp. 1–5, 2017. doi: 10.1109/EI2.2017.8245345.

[87] KUMAR, N., HUSSAIN, I., SINGH, B., et al. "Rapid MPPT for Uniformly and Partial Shaded PV System by Using JayaDE Algorithm in Highly Fluctuating Atmospheric Conditions", *IEEE Transactions on Industrial Informatics*, v. 13, n. 5, pp. 2406–2416, 2017. ISSN: 15513203. doi: 10.1109/TII.2017.2700327.

[88] DAS, T., ROY, R. "Optimal reactive power dispatch using JAYA algorithm", *2018 Emerging Trends in Electronic Devices and Computational Techniques, EDCT 2018*, pp. 1–6, 2018. doi: 10.1109/EDCT.2018.8405071.

[89] SINGH, P., PUREY, P., TITARE, L. S., et al. "Optimal reactive power dispatch for enhancement of static voltage stability using jaya algorithm", *IEEE International Conference on Information, Communication, Instrumentation and Control, ICICIC 2017*, v. 2018-Janua, pp. 1–5, 2018. doi: 10.1109/ICOMICON.2017.8279044.

[90] WARID, W., HIZAM, H., MARIUN, N., et al. "A novel quasi-oppositional jaya algorithm for optimal power flow solution", *2018 International Conference on Computing Sciences and Engineering, ICCSE 2018 - Proceedings*, pp. 1–5, 2018. doi: 10.1109/ICCSE1.2018.8373995.

[91] SAMUEL, O., JAVAID, N., ASLAM, S., et al. "JAYA optimization based energy management controller for smart grid: JAYA optimization based energy management controller", *2018 International Conference on Computing, Mathematics and Engineering Technologies: Invent, Innovate and Integrate for Socioeconomic Development, iCoMET 2018 - Proceedings*, v. 2018-Janua, pp. 1–8, 2018. doi: 10.1109/ICOMET.2018.8346337.

[92] DASH, S. K., RAY, P. K. "Power Quality Improvement Utilizing PV Fed Unified Power Quality Conditioner Based on UV-PI and PR-R Controller", *CPSS Transactions on Power Electronics and Applications*, v. 3, n. 3, pp. 243–253, 2018. ISSN: 2475742X. doi: 10.24295/cpsstpea.2018.00024.

[93] LATHER, J. S., DHILLON, S. "Multi area load frequency control of interconnected power systems using JAYA", *2017 IEEE Electrical Power*

and Energy Conference, EPEC 2017, v. 2017-Octob, pp. 1–6, 2018. doi: 10.1109/EPEC.2017.8286241.

[94] HUANG, C., WANG, L., YEUNG, R. S. C., et al. "A prediction model-guided jaya algorithm for the PV system maximum power point tracking", *IEEE Transactions on Sustainable Energy*, v. 9, n. 1, pp. 45–55, 2018. ISSN: 19493029. doi: 10.1109/TSTE.2017.2714705.

[95] BARAKAT, A. F., EL-SEHIEMY, R. A., ELSAYD, M. I., et al. "An Enhanced Jaya Optimization Algorithm (EJOA) for Solving Multi-Objective ORPD Problem", *Proceedings of 2019 International Conference on Innovative Trends in Computer Engineering, ITCE 2019*, pp. 479–484, February 2019. doi: 10.1109/ITCE.2019.8646363.

[96] BASAK, S., BANERJEE, S. "Incorporation of Solar-Wind Energy in ELD with Thermal Units by Adaptive Jaya Algorithm for Microgrid", *Proceedings of 2nd International Conference on Innovations in Electronics, Signal Processing and Communication, IESC 2019*, v. 0, n. 3, pp. 197–201, 2019. doi: 10.1109/IESPC.2019.8902464.

[97] EI-ELA, A. A. A., MOUWAFI, M. T., SHAMMAH, A. E. S., et al. "Optimal Placement of D-STATCOM in Distribution Systems Using JAYA Algorithm", *2018 20th International Middle East Power Systems Conference, MEPCON 2018 - Proceedings*, pp. 834–838, 2019. doi: 10.1109/MEPCON.2018.8635133.

[98] JAVIDSHARIFI, M., NIKNAM, T., AGHAEI, J., et al. "Probabilistic Model for Microgrids Optimal Energy Management Considering AC Network Constraints", *IEEE Systems Journal*, pp. 1–10, 2019. ISSN: 1932-8184. doi: 10.1109/jsyst.2019.2927437.

[99] ROUTRAY, A., MISTRY, K. D., ARYA, S. "Loss Minimization in a Radial Distribution System with DG Placement Using Jaya Optimization Technique", *1st IEEE International Conference on Sustainable Energy Technologies and Systems, ICSETS 2019*, pp. 336–340, 2019. doi: 10.1109/ICSETS.2019.8745110.

[100] KUMARI, V., NAYAR, S., RAJ, R. S. "Enhanced Automatic generation control of two area power system with Jaya algorithm", *Proceedings of the 3rd International Conference on Electronics and Communication and Aerospace Technology, ICECA 2019*, pp. 439–444, 2019. doi: 10.1109/ICECA.2019.8821982.

[101] PADMANABAN, S., PRIYADARSHI, N., BHASKAR, M. S., et al. "A Hybrid Photovoltaic-Fuel Cell for Grid Integration with Jaya-Based Maximum Power Point Tracking: Experimental Performance Evaluation", *IEEE Access*, v. 7, pp. 82978–82990, 2019. ISSN: 21693536. doi: 10.1109/ACCESS.2019.2924264.

[102] MANZOOR, A., JUDGE, M. A., ALMOGREN, A., et al. "A priori multiobjective self-adaptive multi-population based Jaya algorithm to optimize DErS operations and electrical tasks", *IEEE Access*, v. 8, pp. 181163–181175, 2020. ISSN: 21693536. doi: 10.1109/ACCESS.2020.3028274.

[103] KUMAR, B. S., RASTOGI, A. K., RAJANI, B., et al. "Optimal Solution to Economic Load Dispatch by Modified Jaya Algorithm", *2021 6th International Conference on Recent Trends on Electronics, Information, Communication and Technology, RTEICT 2021*, pp. 348–352, 2021. doi: 10.1109/RTEICT52294.2021.9574009.

[104] UR REHMAN, A., HAFEEZ, G., ALBOGAMY, F. R., et al. "An efficient energy management in smart grid considering demand response program and renewable energy sources", *IEEE Access*, v. 9, pp. 148821–148844, 2021. ISSN: 21693536. doi: 10.1109/ACCESS.2021.3124557.

[105] MOTAMARRI, R., BHOOKYA, N. "JAYA Algorithm Based on Lévy Flight for Global MPPT under Partial Shading in Photovoltaic System", *IEEE Journal of Emerging and Selected Topics in Power Electronics*, v. 9, n. 4, pp. 4979–4991, 2021. ISSN: 21686785. doi: 10.1109/JESTPE.2020.3036405.

[106] DEBOUCHA, H., SHAMS, I., BELAID, S. L., et al. "A fast GMPPT scheme based on Collaborative Swarm Algorithm for partially shaded photovoltaic system", *IEEE Journal of Emerging and Selected Topics in Power Electronics*, v. 9, n. 5, pp. 5571–5580, 2021. ISSN: 21686785. doi: 10.1109/JESTPE.2021.3071732.

[107] HUY ANH, H. P., QUOC KHANH, P., VAN KIEN, C. "Advanced pmsm machine parameter identification using modified jaya algorithm", *Proceedings of 2019 International Conference on System Science and Engineering, ICSSE 2019*, pp. 445–450, 2019. doi: 10.1109/ICSSE.2019.8823434.

[108] YU, K., LIANG, J. J., QU, B. Y., et al. "Parameters identification of photovoltaic models using an improved JAYA optimization algorithm", *Energy Conversion and Management*, v. 150, n. August, pp. 742–753, 2017. ISSN: 01968904. doi: 10.1016/j.enconman.2017.08.063.

[109] BEDEKAR, P. P., KORDE, P. N. "Optimum coordination of overcurrent relays using the modified Jaya algorithm", *2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering, UPCON 2016*, pp. 479–484, 2017. doi: 10.1109/UPCON.2016.7894701.

[110] FOSTER, J. G. L., PEREIRA, R. R., GONZATTI, R. B., et al. "A Review of FCS-MPC in Multilevel Converters Applied to Active Power Filters". In: *2019 IEEE 15th Brazilian Power Electronics Conference and 5th IEEE Southern Power Electronics Conference (COBEP/SPEC)*, pp. 1–6. IEEE, dec 2019. ISBN: 978-1-7281-4180-0. doi: 10.1109/COBEP/SPEC44138.2019.9065398.

[111] CAICEDO, J., AREDES, M., WATANABE, E. H. "Frequency analysis behavior of hysteresis current control in VSI", *2013 Power Electronics and Power Quality Applications, PEPQA 2013 - Proceedings*, v. 2, n. 3, 2013. doi: 10.1109/PEPQA.2013.6614949.

Appendix A

Disturbance Rejection

During the development of this thesis, the disturbance rejection of Jaya-MPC was an object of study in the stage of $MATLAB^{TM}$ simulation to design it. High disturbance rejection performance is a common characteristic of MPC in power electronics because of their fast dynamical response and the fact the predictor model includes the measurement of these disturbances. The system of Figure 4.2 was modeled in $Simulink^{TM}$ to analyze a set of three types of disturbances: ac grid voltage harmonics, ac grid voltage unbalancing — due to negative sequence components—, and dc-link voltage variations.

Figure A.1 shows the simulation setup, which is composed of three blocks: the current reference generator based on the pq-theory, the voltage-disturbances generator, and the three-phase converter system that contains the model of the plant, and the Jaya-MPC block, which calls the Jaya-MPC algorithm and provides the optimal modulation index to the PWM-block. After the end of the simulation, a $MATLAB^{TM}$ algorithm generates the graphics. Table A.1 shows the simulation parameters.

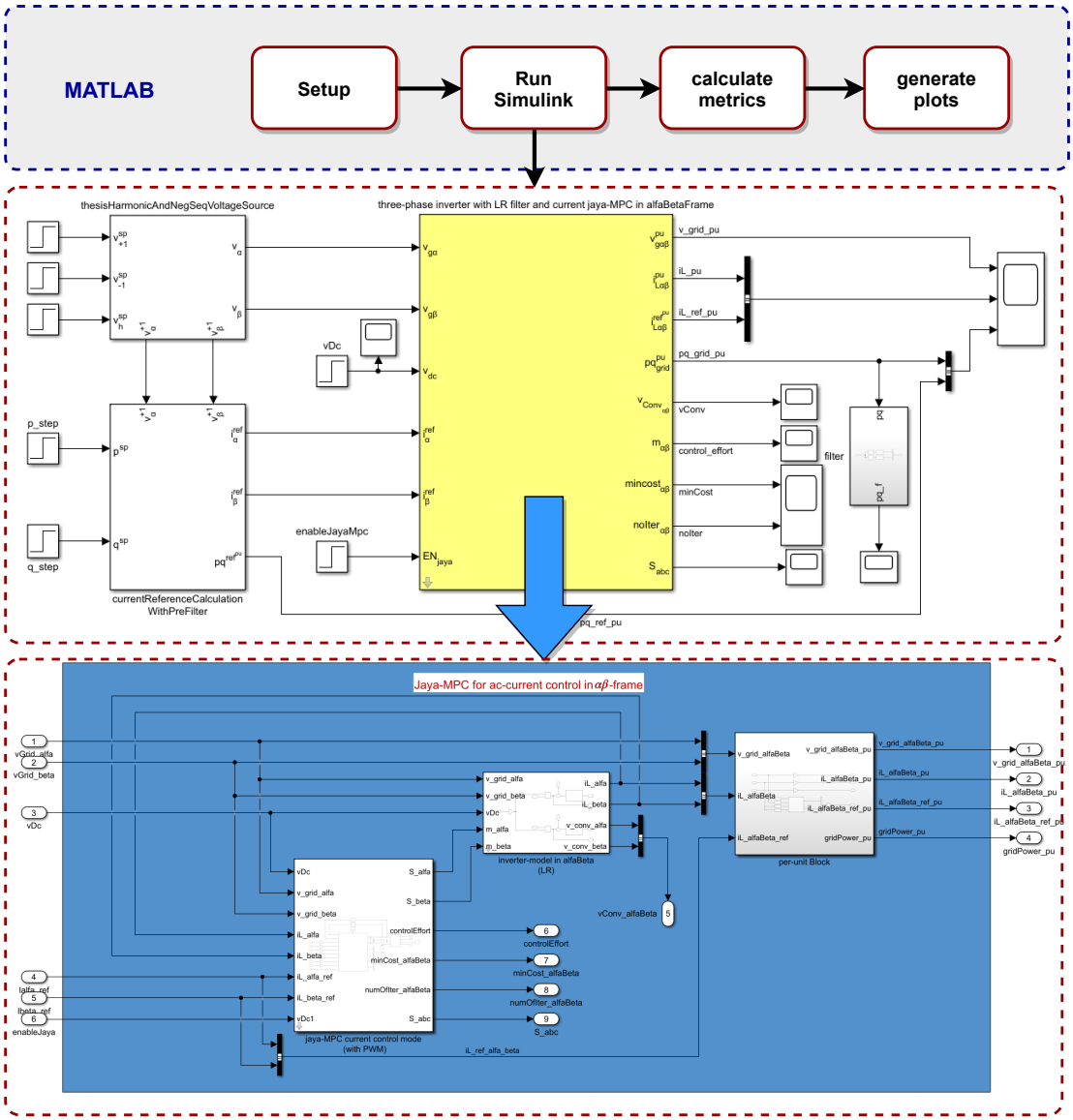


Figure A.1: Exploded view of the system modeled in $Simulink^{TM}$ to analyze the Jaya-MPC disturbance rejection.

Table A.1: Simulation parameters for the time-domain analyses.

Nominal Electrical Parameters	
$V_{rms,LL}$	220 V
V_{dc}	450 V
S	10 KA
f	60 Hz
Filter Parameters	
L	2.03 mH
R	30.6 mΩ
Jaya-MPC Parameters	
N	8
tol	10^{-4}
Switching Frequency (F_{sw})	5940 Hz
Sampling Period ($\frac{1}{F_{sw}}$)	168.35 μs
weighting-factors strategy	correlated-adaptive
initial weight-factors	($r_1 = 0.33, r_2 = 0.33$)
Simulation Parameters	
simulation time step	1.68 us
disturbances occur at	0.05 s

A.1 Disturbance Rejection Analysis

Besides the fast reference-tracking response, converters controllers must present a disturbance-rejection capability when dealing with grid-harmonics or negative-sequence disturbances in the grid voltage. The following results address these issues and demonstrate the effectiveness of Jaya-MPC in keeping the ac-currents regulated with a high-power quality even when the grid voltage presents these disturbances.

A.1.1 Rejection of Grid Harmonics

Figure A.2 shows the electrical quantities of the converter in a scenario with the presence of grid harmonics. In (a), it presents the grid voltages in the $\alpha\beta$-frame with a harmonic-content disturbance starting at the instant of 0.05 s. This disturbance results in a voltage THD equal to 6.8%. Note that, in (b), the ac-currents keep regulated with a high power quality due to the Jaya-MPC fast-response, leading to a THD no greater than 1%. In (c), the active and reactive power start to present oscillations ($\tilde{p}_{conv}, \tilde{q}_{conv}$) because of the low-frequency harmonics in the grid voltage.

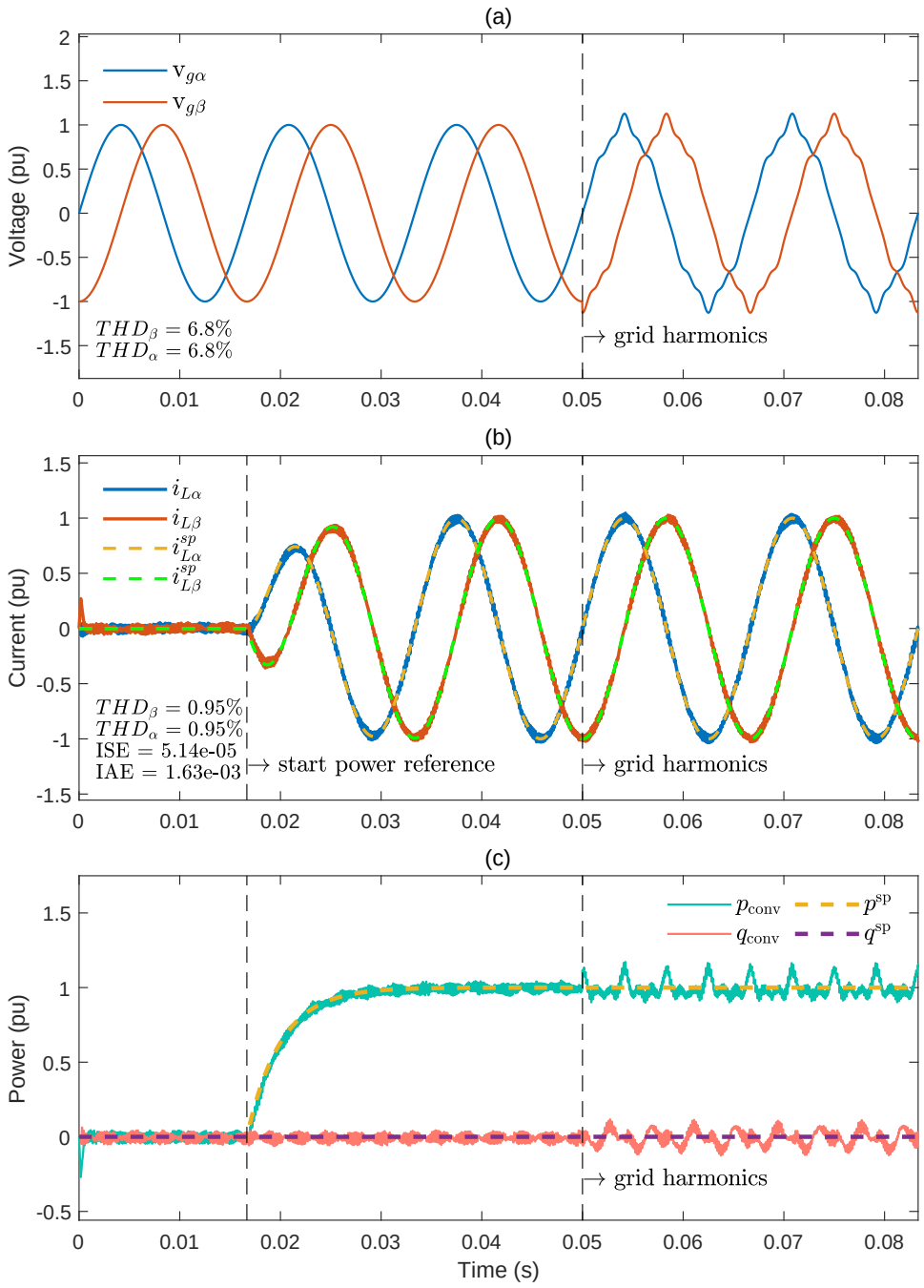


Figure A.2: Jaya-MPC response under grid voltage harmonics: (a) waveforms of the grid-voltage with a harmonic disturbance in $0.05s$ ($THD = 6.8\%$); (b) waveforms of the ac-currents with lower THD; (c) active and reactive power with oscillation due to the grid voltage harmonics.

In the same scenario of Figure A.2, Figure A.3 shows the control signals of Jaya-MPC. In (a), it depicts the modulation index generated by the Jaya algorithm fast responding to the grid-voltage disturbance and adding the respective harmonic components to the converter voltage to block the path for ac-current harmonics.

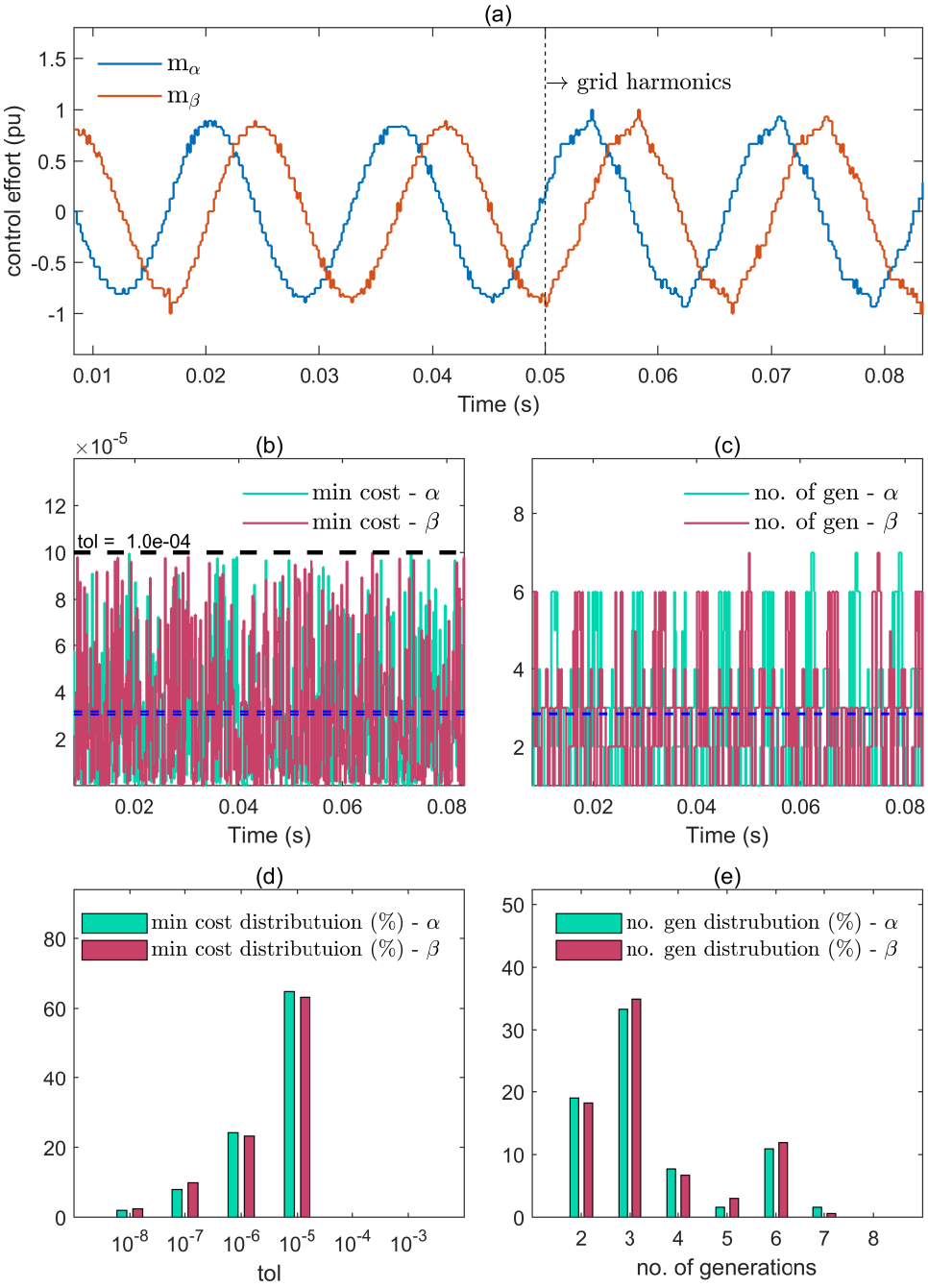


Figure A.3: Jaya-MPC response under grid voltage harmonics: (a) modulation index calculated by the Jaya algorithm; (b) optimal cost profile for each sampling instant; (c) number-of-generations that Jaya-MPC took to return the optimal control-action for each sampling instant; (d) distribution of the optimal-cost profile; (e) distribution of the number-of-generations profile.

This effort leads to an increase in the instantaneous minimum cost (b) and the number of generations (c) that Jaya-MPC achieves and takes to return the optimal modulation index presented in (a). But even with this increase, the average minimum cost and the average number of generations (blue dashed lines) remain below 10^{-4} and 4, respectively; the distribution of these two metrics is presented in (d) and (e) confirms this conclusion.

A.1.2 Rejection of Negative Sequence Voltage

Figure A.4 shows the electrical quantities of the converter in a scenario with the presence of a negative-sequence disturbance in the grid voltage; in (a), it presents the grid voltages in the $\alpha\beta$-frame with a 10% negative-sequence component starting at the instant of 0.05 s. Note that, in (b), the ac-currents keep regulated with a high power quality due to the Jaya-MPC fast-response, leading to a THD lower than 1%. In (c), the active and reactive power start to present oscillated components ($\tilde{p}_{conv}, \tilde{q}_{conv}$) because of the negative-sequence component in the grid-voltage.

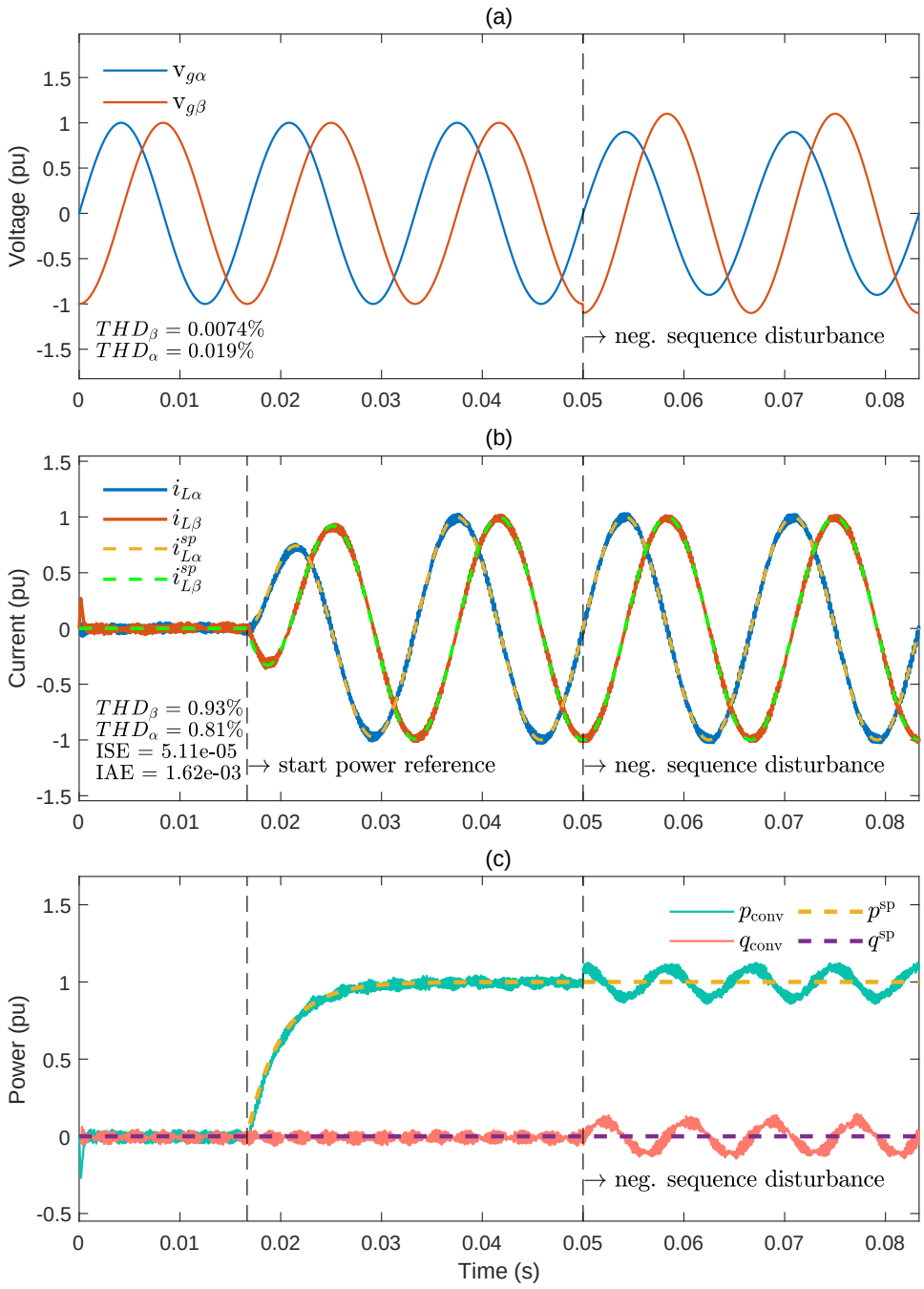


Figure A.4: Jaya-MPC response under a negative-sequence disturbance in the grid voltage: (a) waveforms of the grid-voltage with a negative-sequence disturbance in $0.05s$; (b) waveforms of the balanced ac-currents with lower THD; (c) active and reactive power with oscillation due to the grid voltage negative sequence components.

In the same scenario of Figure A.4, Figure A.5 shows the control signals of Jaya-MPC; in (a), it depicts the modulation index generated by the Jaya algorithm fast responding to the grid-voltage disturbance and adding the negative-sequence component to the converter voltage to block the path for negative-sequence currents.

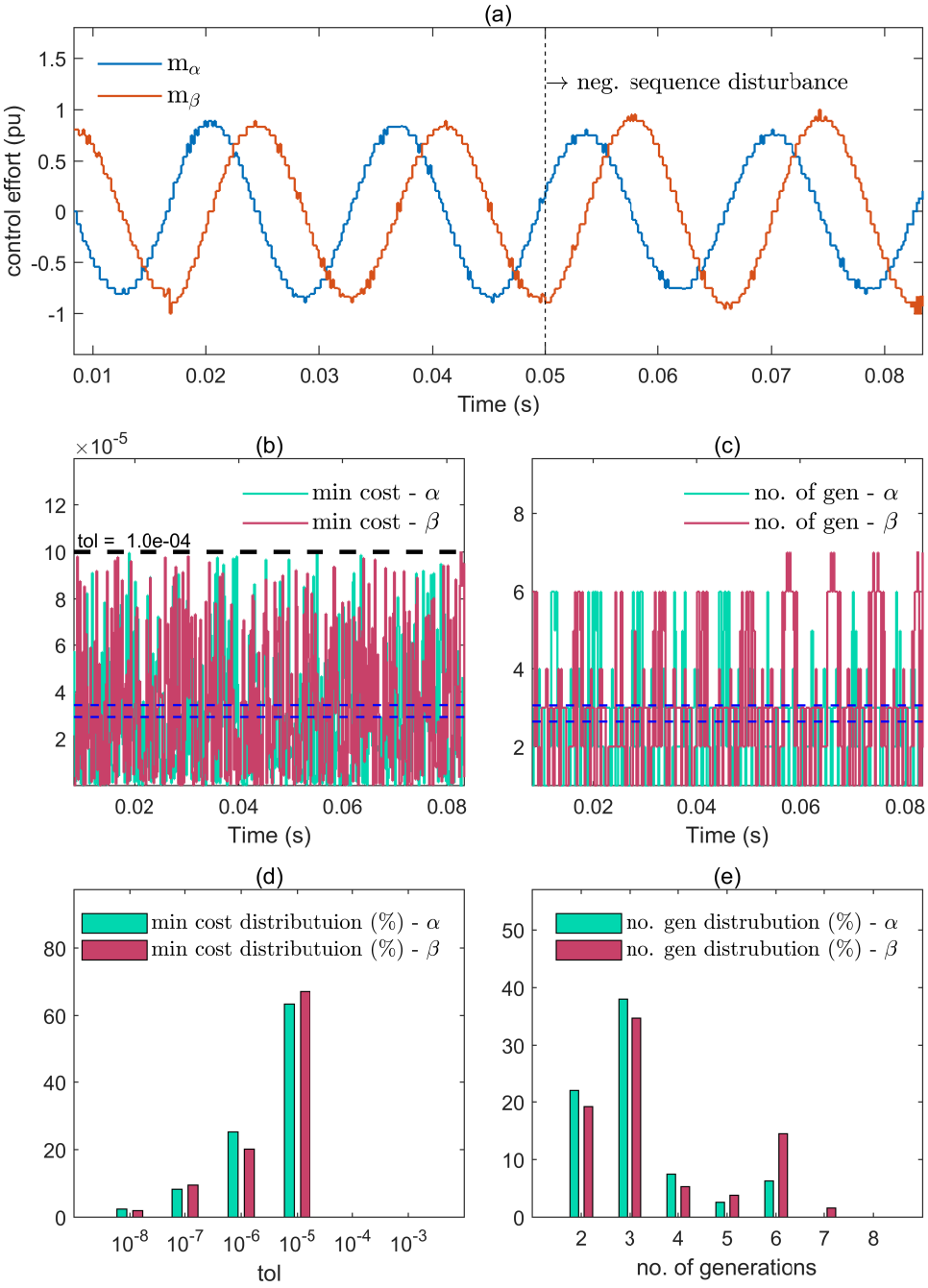


Figure A.5: Jaya-MPC response under a negative-sequence disturbance in the grid voltage: (a) modulation index calculated by the Jaya algorithm; (b) optimal cost profile for each sampling instant; (c) number-of-generations that Jaya-MPC took to return the optimal control-action for each sampling instant; (d) distribution of the optimal-cost profile; (e) distribution of the number-of-generations profile.

This effort does not lead to an increase in the instantaneous minimum cost (b) but raises the number of generations (c) that Jaya-MPC takes to return the optimal modulation index of the β-component and reduces the one of the α-component.

This characteristic results in an average number of generations (blue dashed lines in (c)) slightly inferior to 4 for the α-component and slightly superior to 4 for the β-component; the distribution of these two metrics presented in (d) and (e) confirms these conclusions.

A.1.3 Rejection of Dc-Link Disturbance

The following disturbance-rejection analysis deals with dc-link voltage variation — which may occur during the operation of power converters — and must be addressed by the controller to not affect the system dynamics at the ac-side of the converter. Figure A.6 shows the electrical quantities of the converter in a scenario with the presence a step of 10 % in the dc-link voltage (a).

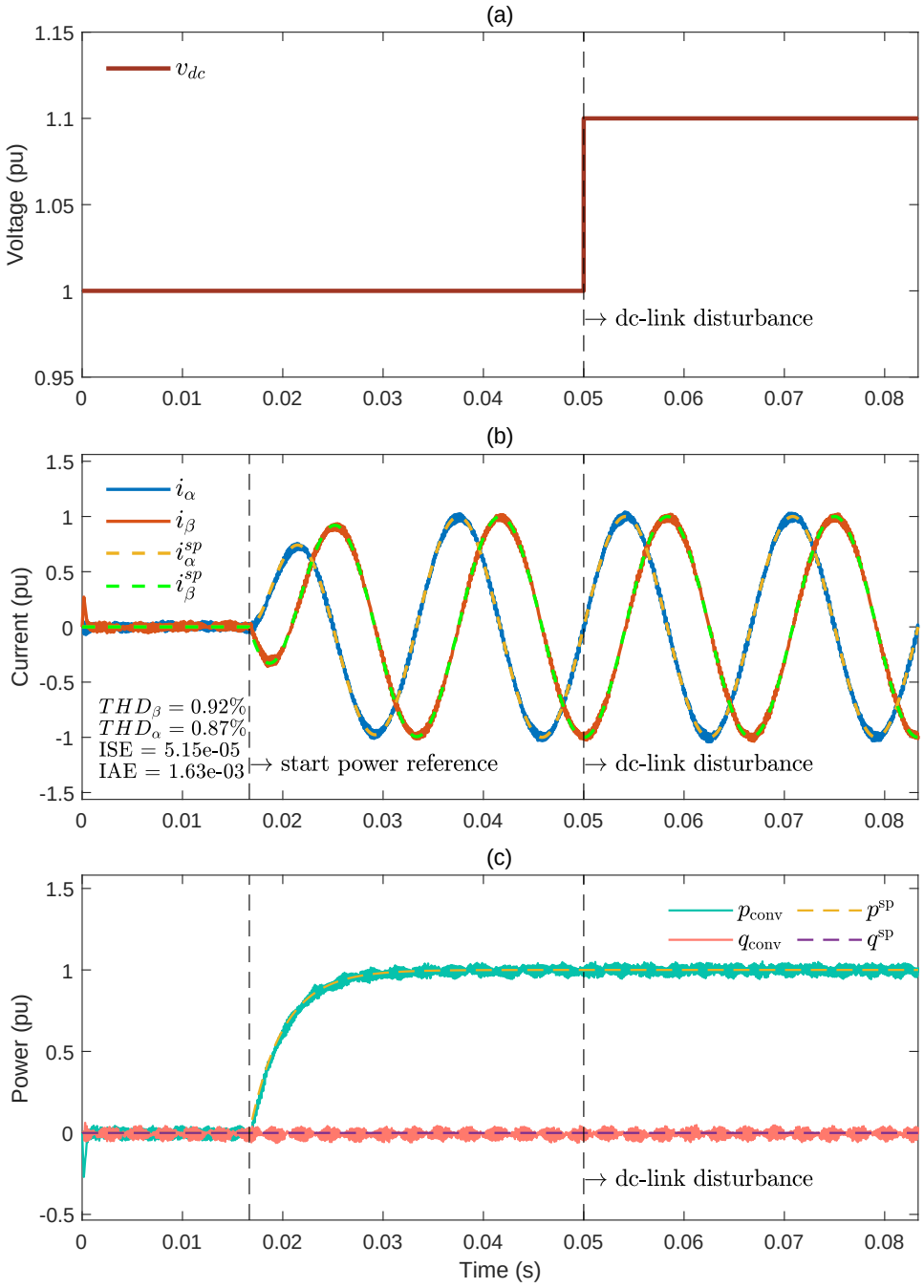


Figure A.6: Jaya-MPC response under a 10% step of the dc-link voltage: (a) dc-link voltage with a positive-step disturbance in $0.05s$; (b) waveforms of the ac-currents with lower THD; (c) active and reactive power with no oscillations.

The Jaya-MPC rejection capability ensures that this dc voltage deviation leads to no ac-current disturbance (b), keeping with the low values of ISE e IAE. The active and reactive power also remains constant during the disturbance (see (c)), confirming that the regulation keeps unchanged.

Note that the Jaya-MPC reduces its control-efforts to compensate for the dc-link voltage increase (see Figure A.7(a)). This compensation presents no effect in the instantaneous minimum cost achieved by Jaya-MPC (see Figure A.7(b)), but reduces the peaks of the instantaneous number-of-generations (see Figure A.7(c)). The distributions of Figure A.7 (d) and (e) confirm these findings.

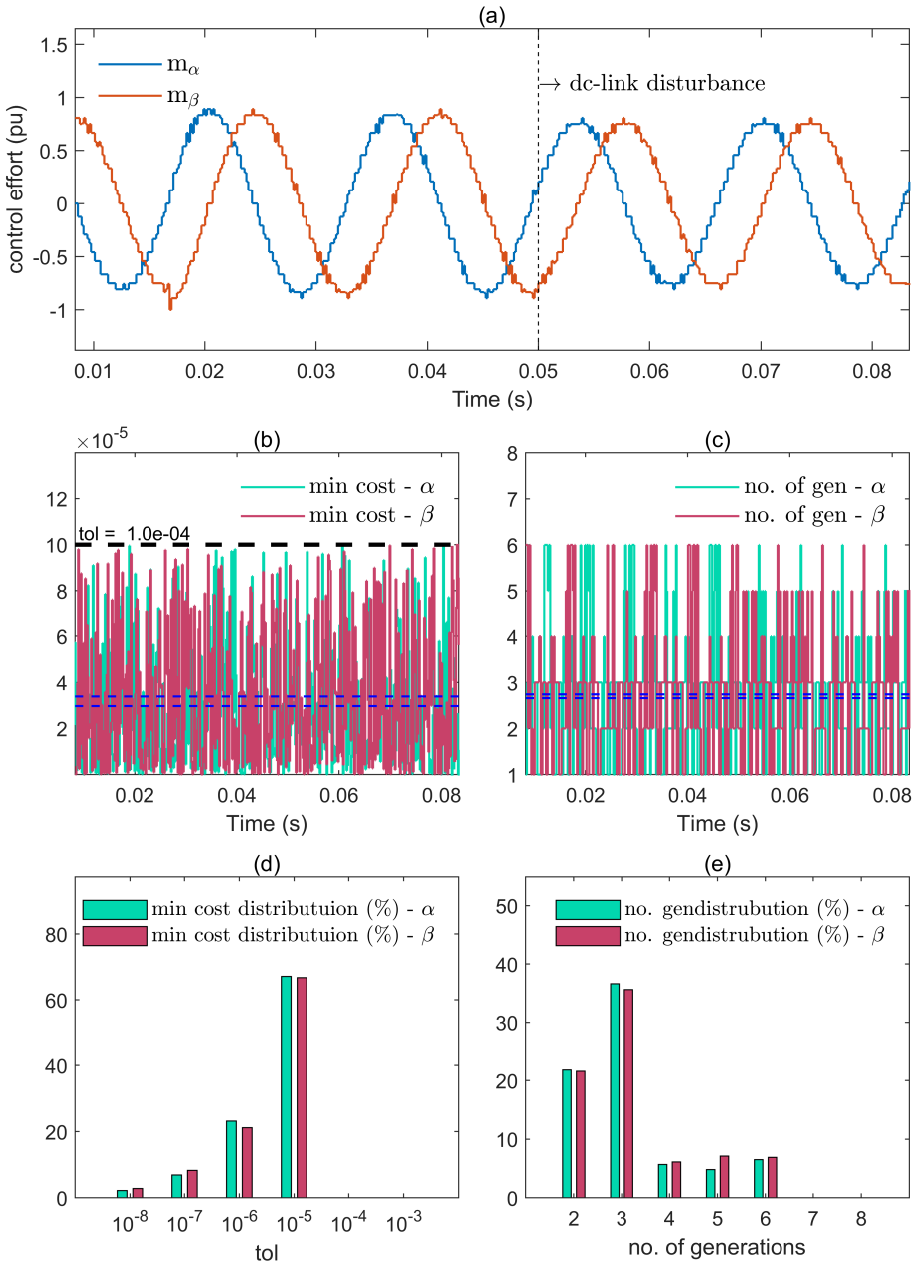


Figure A.7: Jaya-MPC response under a 10% step of the dc-link voltage: (a) modulation index calculated by the Jaya algorithm; (b) optimal cost profile for each sampling instant; (c) number-of-generations that Jaya-MPC took to return the optimal control-action for each sampling instant; (d) distribution of the optimal-cost profile; (e) distribution of the number-of-generations profile.

Figure A.8 shows the electrical quantities of the converter in a scenario with the presence of a step reduction of 10 % in the dc-link voltage (a). The Jaya-MPC rejection capability ensures that this dc-voltage deviation provokes no ac-current disturbance (b), keeping them with low values of ISE e IAE; the active and reactive power also remains constant during the disturbance (see (c)).

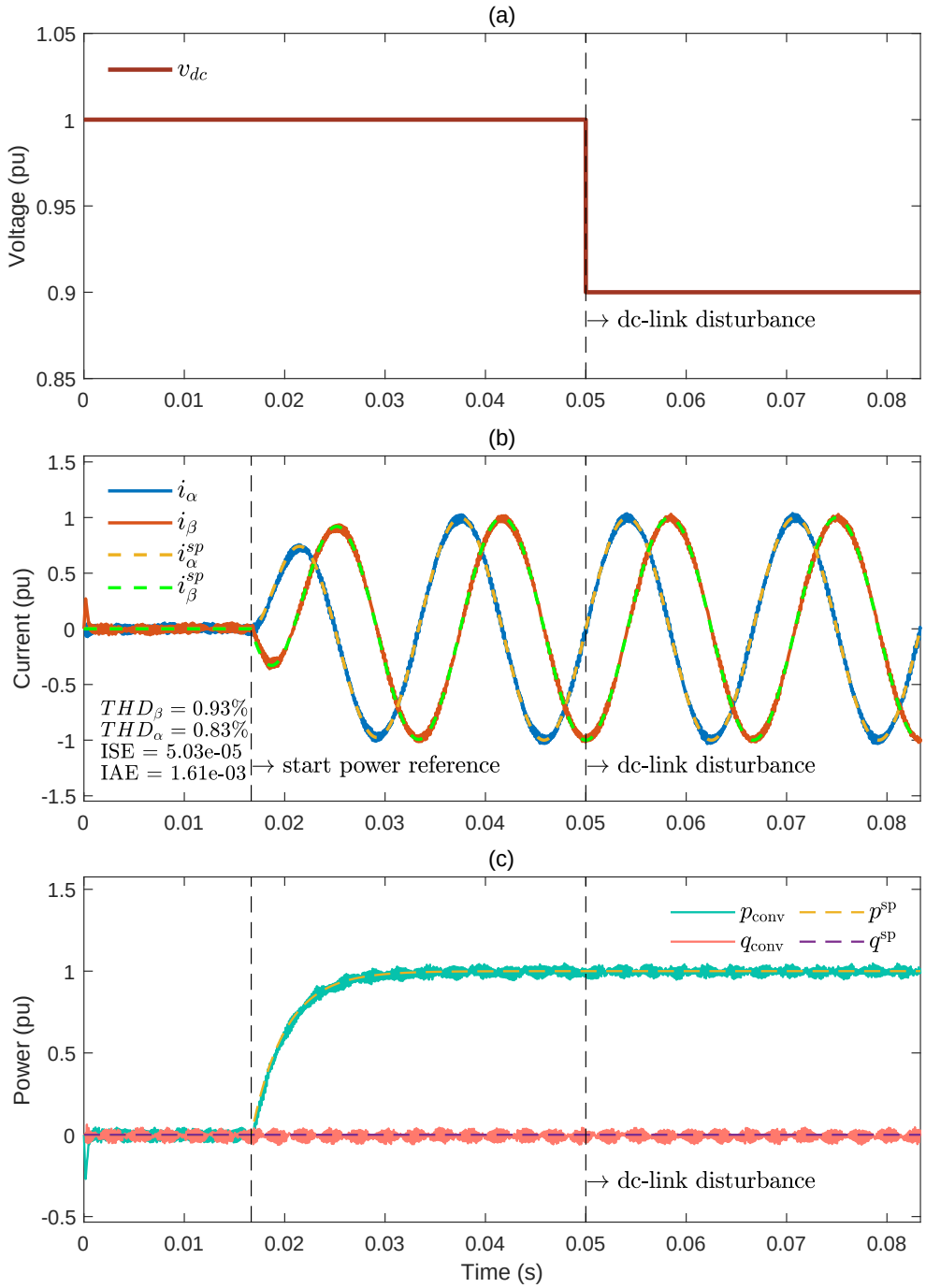


Figure A.8: Jaya-MPC response under a 10% negative-step of the dc-link voltage: (a) dc-link voltage with a negative-step disturbance in $0.05s$; (b) waveforms of the ac-currents with lower THD; (c) active and reactive power with no oscillations.

The Jaya-MPC increases its control-efforts to compensate for the dc-link voltage decrease (see Figure A.9(a)). This compensation presents a slightly increase in a few peaks of the instantaneous minimum cost achieved by Jaya-MPC (see Figure A.9(b)), and also increases the instantaneous number-of-generations (see Figure A.9(c)). The distributions of Figure A.9 (d) and (e) confirm these findings.

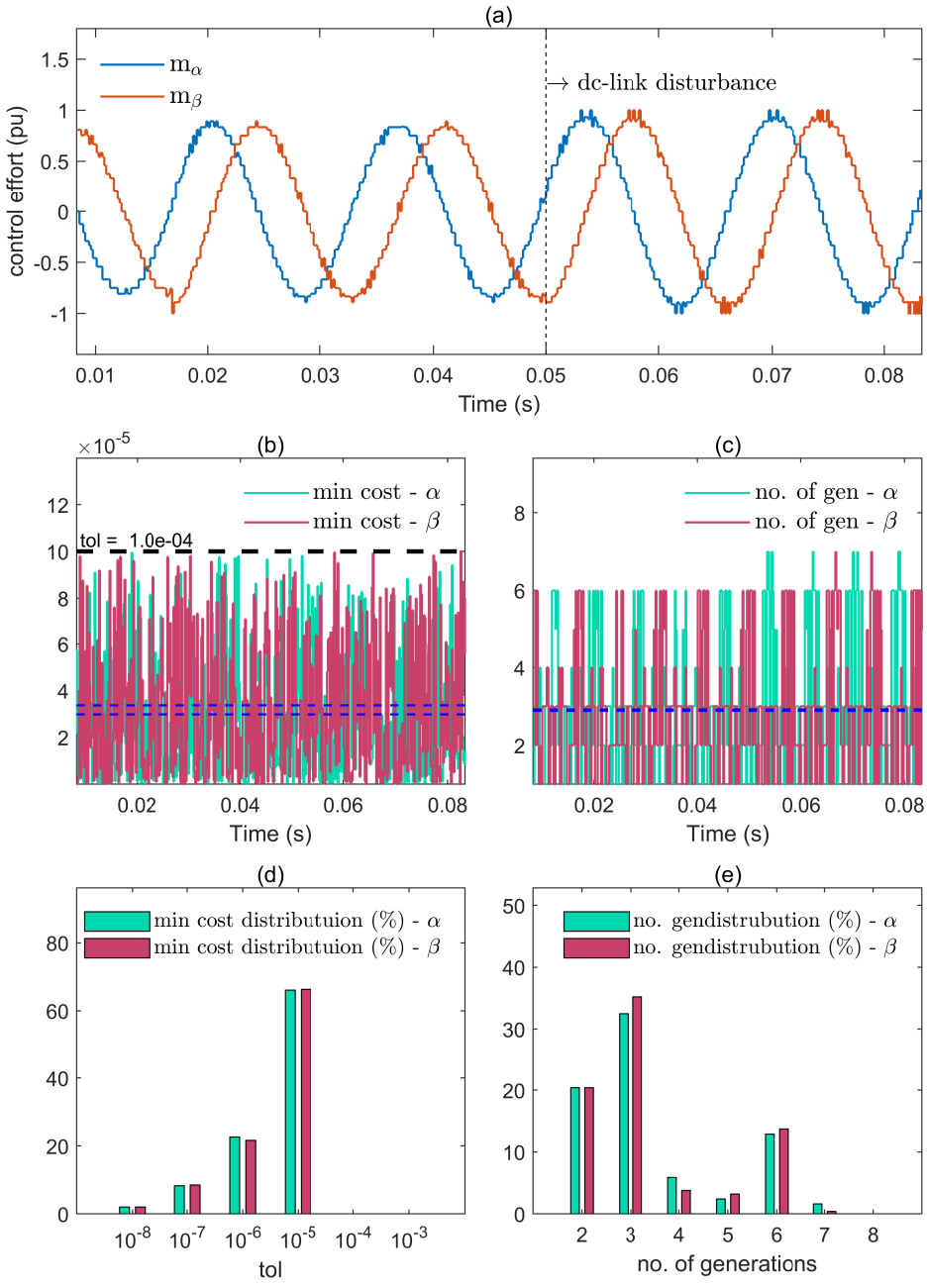


Figure A.9: Jaya-MPC response under a 10% negative-step of the dc-link voltage: (a) modulation index calculated by the Jaya algorithm; (b) optimal cost profile for each sampling instant; (c) number-of-generations that Jaya-MPC took to return the optimal control-action for each sampling instant; (d) distribution of the optimal-cost profile; (e) distribution of the number-of-generations profile.

A.2 Model Parameter Sensibility

Figure A.10 compares the model parameter sensibility of both Jaya-MPC and FCS-MPC facing a filter parameter uncertainty ΔL. While Jaya-MPC presents a low sensibility, maintaining THD below 5%, the FCS-MPC can only achieve this level of robustness with greater values of sampling frequency. This first result strongly suggests that Jaya-MPC is a more robust approach than FCS-MPC. However, future research can deepen these findings by analyzing how the sampling frequency affects both the parameter sensibility and switching frequency profile produced by FCS-MPC.

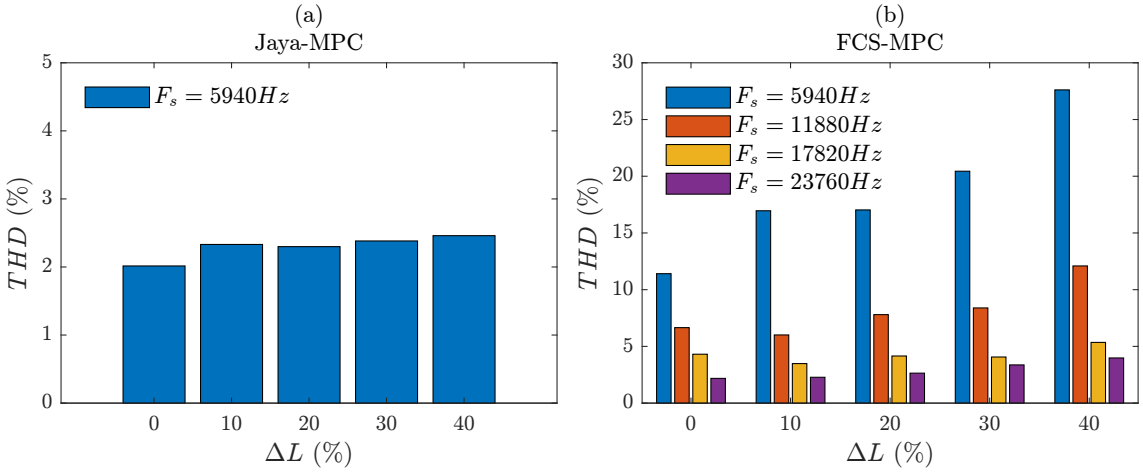


Figure A.10: Model parameter sensibility: THD of Jaya-MPC (a) and FCS-MPC (b) when subject to a model parameter error ΔL.

Appendix B

Random-weights on Jaya-MPC

Figure B.1 shows the statistical analysis of Jaya-MPC using the original Jaya algorithm weights with uniform distribution. This analysis ran 100 experiments and calculated the statistics for the THD — the blue 'x' is a discarded outlier. The THD ranges from 5.5% to 6.4%, with an average value slightly greater than 6.

As the uniform distribution produces random values with an expected value of 0.5, this result presents an average THD close to the value produced by the correlated fixed weights strategy of Figure 5.8 (c) with $r_1 = r_2 = 0.5$. This result confirms that the weighting strategy proposed in this work suits better the scope of the application of Jaya-MPC, but future studies can determine a better statistical distribution for the Jaya-MPC weights.

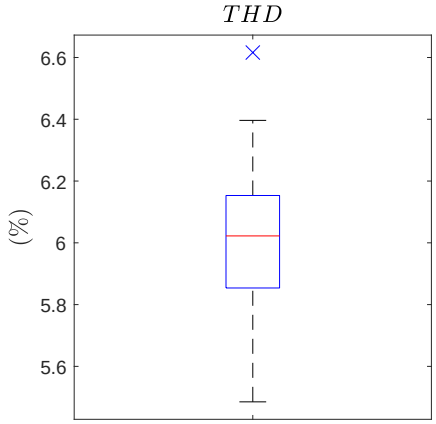


Figure B.1: THD of Jaya-MPC using the original random weights in the Jaya algorithm with uniform distribution.

Appendix C

Switching Transitions in The Cost Function

One of the first approaches to deal with the variable switching frequency in FCS-MPC was to add the switching states transition ($S_n^p - S_n^k$) as a term of the cost function [8, 9], as in Equation (C.1):

$$J = ||i_{L_\alpha}^{sp} - i_{L_\alpha}^{p}||^2 + ||i_{L_\beta}^{sp} - i_{L_\beta}^{p}||^2 + \lambda \sum_n |S_n^p - S_n^k| \tag{C.1}$$

This approach aims to concentrate the switching frequency components within a smaller range by configuring the weight parameter λ, but now, it is possible to quantitatively determine the contribution of this term in the switching frequency spread by analyzing the effect of λ in the TFS.

Figure C.1 (a) shows that neither THD nor TFS benefits from the use of this method; in fact, the TFS is worse when λ differs from zero, and, in (b), one can observe this method contributes to reducing both average switching frequency and dominant frequency.

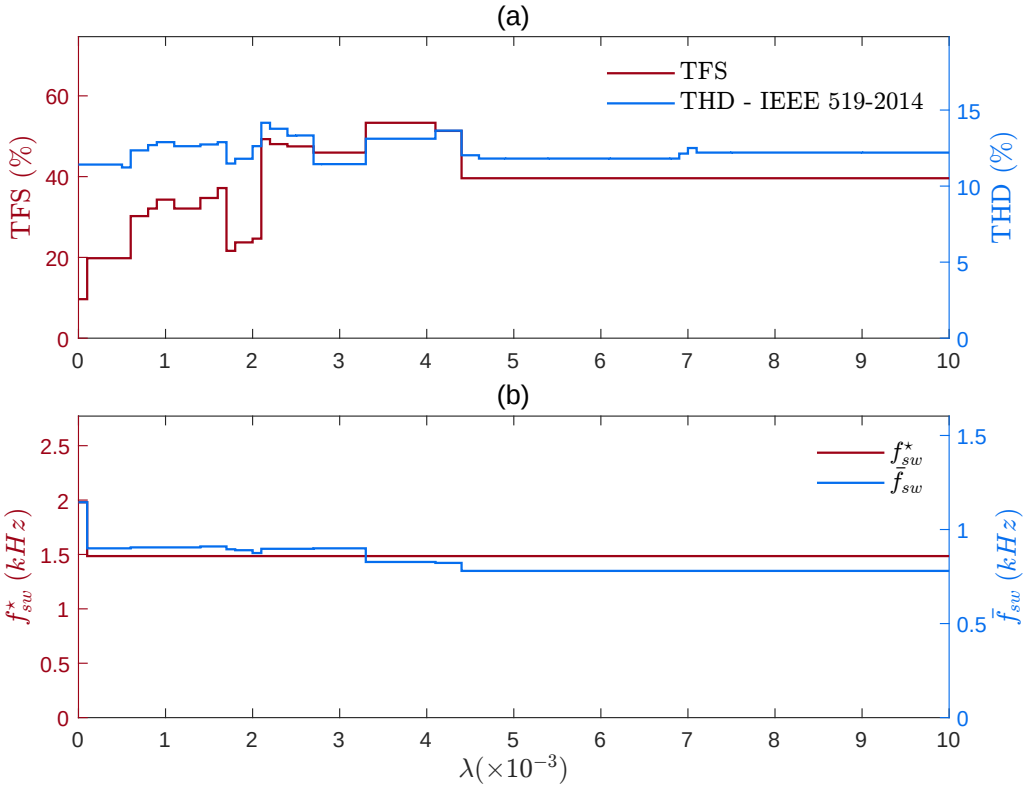


Figure C.1: Effect of λ using FCS-MPC with cost function (C.1) and sampling frequency equal to 5940 Hz ($F_s = 99 \times 60\,Hz$): (a) TFS and THD; (b) $f^{\star}_{sw}$ and $\bar{f}_{sw}$.

On the other hand, in the case which $F_s = 29460Hz$ — where the TFS peak is highest in analyses of Chapter 6 (see Figure 6.6) —, Figure C.2 (a) shows that λ impacts significant in TFS reduction. However, THD increases because it is highly correlated with the average switching frequency, i.e., as the value of $\bar{f}_{sw}$ decreases, the converter makes fewer switching transitions, leading to greater ripples, therefore, jeopardizing power quality.

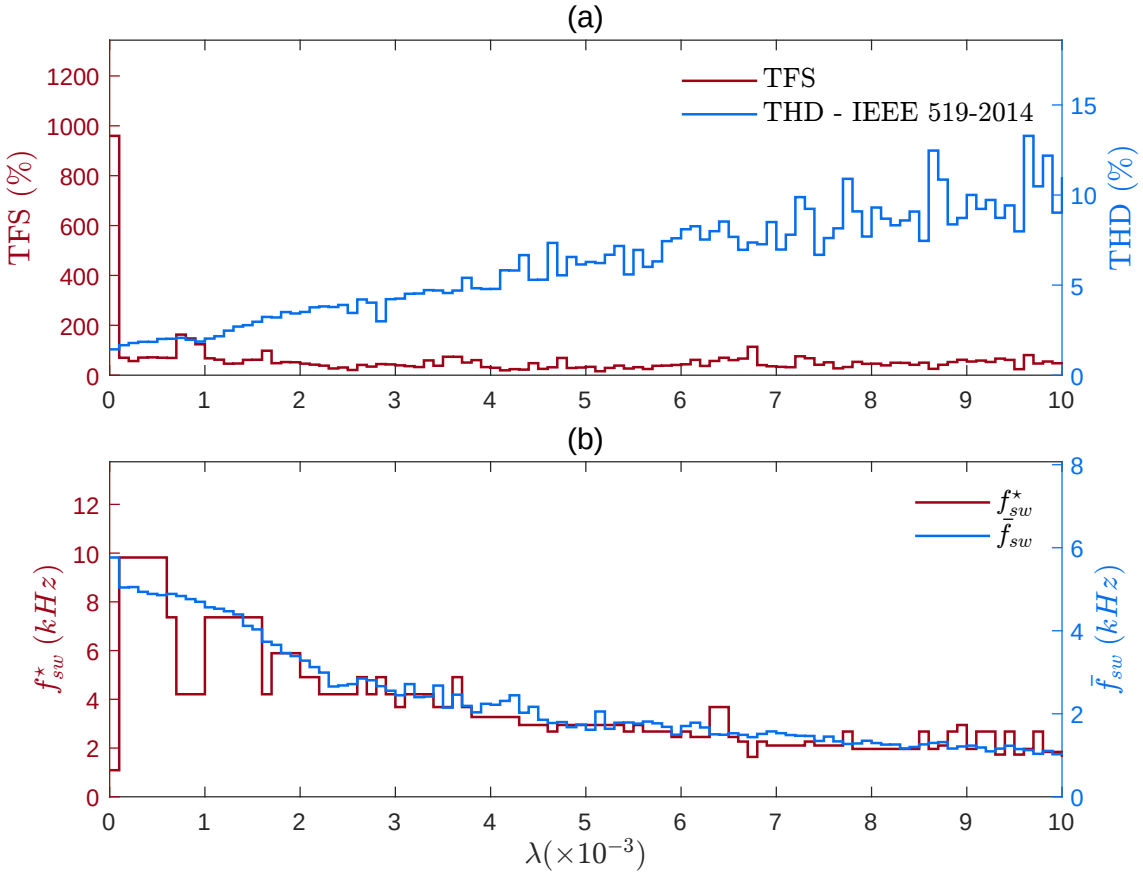


Figure C.2: Effect of λ using FCS-MPC with cost function (C.1) and sampling frequency equal to 29460 Hz ($F_s = 491 \times 60\ Hz$): (a) TFS and THD; (b) $f^{\star}_{sw}$ and $\bar{f}_{sw}$.

These findings demonstrate that the technique of adding the switching transitions to the cost function does not ensure the expected behavior of reducing the switching frequency variability. Again, it depends on the sampling period since FCS-MPC produces a highly nonlinear and not well-defined switching frequency profile. But, with the TFS metric, future studies can analyze each method in the literature to assess their contribution to the reduction of the switching frequency spread.