National Technical University of Athens

School of Electrical and Computer Engineering

Division of Electric Power

Model Predictive Control Strategies for Power Electronics Converters and AC Drives

PhD Thesis

Petros Karamanakos

Athens, July 2013

National Technical University of Athens

School of Electrical and Computer Engineering

Division of Electric Power

Model Predictive Control Strategies for Power Electronics Converters and AC Drives

PhD Thesis

Petros Karamanakos

Three-Member Consultant Committee: Stefanos Manias
Ralph Kennel
Antonios Kladas

Approved from the examination committee on July 24, 2013

.....	.....	.....
Stefanos Manias	Ralph Kennel	Antonios Kladas
Prof. NTUA	Prof. TUM	Prof. NTUA
.....	.....	.....
Tryfon Koussiouris	Stavros Papathanassiou	Nikolaos Maratos
Prof. NTUA	Assoc. Prof. NTUA	Prof. NTUA

.....
Costas Tzafestas
Assist. Prof. NTUA

Athens, July 2013

Petros Karamanakos
Dipl. Electr. & Comput. Eng. NTUA

The greatest challenge to any thinker
is stating the problem in a way that will allow a solution.

Bertrand Russell (1872 - 1970)

Acknowledgments

During my graduate years, as my work evolved, I have accumulated much intellectual debt to the individuals who fostered my Ph.D. career. In the following few lines I am trying to express my deep gratitude to all these people. Nonetheless, in almost every list the human brain tends to focus on items at the beginning and end. Since, written languages are linear, obviously there will be a first person, a last person and many others in between. I would like to mention that this is not because I see these individuals in a hierarchy of values. Thus, I would like to kindly ask from the reader to pay attention not only to the people at the end points, but also to those in the middle.

My deepest gratitude goes to my supervisor Prof. Stefanos Manias for his continuous support, fruitful discussions and life advice. Without his broad experience and right-on advice, I would not have been able to face the challenges of the academic research. I am also very grateful for giving me the opportunity to work on the research topics of my interests, and to collaborate with the people I wanted to.

I am grateful to Prof. Ralph Kennel from Technische Universität München, Munich, Germany, for providing me with access to the laboratory infrastructure in Munich, and for giving me the opportunity to work, even for a while, in a pleasing environment. Furthermore, he contributed with constructive comments on issues related to the present work.

I also would like to thank Profs. Antonios Kladas, Tryfon Koussiouris, Nikolaos Maratos, Assoc. Prof. Stavros Papathanassiou, and Assist. Prof. Costas Tzafestas for taking up the co-examination, as well as for their time and feedback.

During the first years of my Ph.D. I was lucky enough to work with Dr. Georgios Papafotiou, ABB Corporate Research, Baden-Dättwil, Switzerland. Georgios helped me significantly with all my questions related to control, optimization, and the basic concepts of model predictive control. His patience, and feedback established my research direction and improved my research skills.

I am most grateful to Dr. Tobias Geyer, ABB Corporate Research, Baden-Dättwil,

Switzerland, since he is clearly the person to whom I owe the most. His patience and endurance in explaining many of the research stuff, as well as some of the more obscure things were most helpful. His deep insight in the fields of optimal control and power electronics, combined with our creative discussions, have made this collaboration particularly productive.

Many results contained in this thesis are the outcome of fruitful and long discussions with Ph.D. candidate, Dipl.-Ing. Peter Stolze, Technische Universität München, Munich, Germany. Our close collaboration was really enjoyable and for that I thank him deeply; his enthusiasm and critical attitude triggered me to develop many of the ideas presented here. Thanks for existing my friend.

I am also very glad to have had the opportunity to work closely with Drs. Nikolaos Oikonomou and Frederick Kieferndorf, when I joined ABB Corporate Research, Baden-Dättwil, Switzerland. They shared their wide knowledge on medium voltage drives with me, and spent a great deal of time solving all my questions. Nikolaos helped me significantly to improve my critical thinking, as well as my ability to write to a target audience. I consider myself privileged to profit from the daily interaction with him. Rick was always helpful, supportive, comprehensive, and willing to discuss any type of issues with me; each simple conversation with him was pure gold. For all these I thank both from the bottom of my heart.

Moreover, I thank all the graduate students, Ph.D. candidates, and administrative and technical staff with whom I spent my research years in the Electrical Machines and Power Electronics Lab in NTUA. I had the chance to meet many talented people; I wish all the best to all of them. Furthermore, thanks go to my two students, Georgios Patsakis and Ilias Alevras, for our excellent cooperation, and for their willingness to learn as many things as possible.

Finally, I wish to thank my parents, Argyro and Panagiotis, and my sister, Eleni, for their constant support. Knowing they are always there when I need them is very comforting. I have no words to describe my feelings. To them I dedicate my entire thesis.

Abstract

This thesis focuses on model predictive control (MPC) strategies for power electronics converters and ac drives. Due to their switching nature, power electronic systems constitute nonlinear systems with multiple inputs and outputs, and subject to constraints (e.g. the duty cycle should be limited between zero and one). Over the years many control strategies have been proposed that are mainly based on the conventional proportional-integral-derivative (PID) controller combined with nonlinear techniques, such as pulse width modulation (PWM). However, PID controllers are ideally suited to linear, single-input, single-output (SISO), unconstrained control problems. Moreover, controllers of this type are usually tuned to achieve satisfactory performance only in a narrow operating range. Therefore, the problems associated with many power electronics applications and their closed-loop performance still poses theoretical and practical challenges.

A control algorithm that has recently been gaining popularity in the field of power electronics is MPC. MPC is a control strategy that was developed as an alternative strategy to the conventional PID control. Its success is based on the fact that it uses a mathematical model of the plant, which allows the controller to predict the impact of its control actions. Furthermore, MPC is capable of handling complex and nonlinear dynamics, while several design criteria (constraints) can be explicitly included in a simple and effective manner. By imposing constraints on the variables of concern the plant is able to operate at its physical limits without violating them. Thus, the most favorable operation can be obtained, while the operational limits of the plant are fully respected. Hence, thanks to all these advantageous features, MPC has attracted the interest and attention of research and academic communities. Furthermore, the advent of immensely powerful microprocessors with increased computational capabilities enabled its application in the field of power electronics with significant success.

This thesis is divided into two parts. In the first part the key notions behind MPC are presented, including the concepts of optimization, optimal control, and receding horizon policy. In addition, a brief introduction to the modeling of hybrid systems as hybrid

automata is included. Finally, the notion of enumeration strategy is introduced.

The second part is devoted to applications of MPC in the field of power electronics. It consists of three chapters, each of which refers to a different application. More specifically, Chapter 3 is devoted to dc-dc boost converters, Chapter 4 to cascaded H-bridge (CHB) multilevel rectifiers, and Chapter 5 to ac drives.

Chapter 3 presents two MPC approaches for dc-dc boost converters. A discrete-time switched nonlinear (hybrid) model of the converter is derived, which captures both the continuous and the discontinuous conduction mode. The controller synthesis is achieved by formulating an objective function that is to be minimized subject to the model dynamics. In the first approach, MPC is implemented as a current-mode controller. Two control loops are employed, with the inner loop being designed in the framework of MPC. Two different objective functions are formulated and investigated. The control objective, i.e. the regulation of the current to its reference, is achieved by directly manipulating the switch, thus a modulator is not required. The second proposed strategy, utilized as a voltage-mode controller, achieves regulation of the output voltage to its reference, without requiring a subsequent current control loop. Furthermore, for both approaches, a state estimation scheme is implemented that addresses load uncertainties and model mismatches.

In Chapter 4 an MPC strategy is adapted to the CHB multilevel rectifier. The proposed control scheme aims to keep the sinusoidal input current in phase with the supply voltage, and to achieve independent voltage regulation of the H-bridge cells. To do so, the switches are directly manipulated without the need of a modulator. Furthermore, since all the possible switching combinations are taken into account, the controller exhibits favorable performance not only under nominal conditions, but also under asymmetrical voltage potentials and unbalanced loads. Finally, a short horizon is employed in order to ensure robustness; in this way the required computational effort remains reasonable, making it possible to implement the algorithm in a real-time system.

In Chapter 5 an approach to include a variable switching time point into predictive torque control (PTC) is introduced. In PTC the switching frequency is limited by the sampling frequency; its theoretical maximum value is half the sampling frequency. However, in reality the switching frequency is lower than this value, and thus high current and torque ripples occur compared to modulator-based control methods. In order to overcome this an optimization problem is formulated and solved in real-time. Thereby, apart from the regulation of the torque and the stator flux magnitude to their references, an additional control objective should be met: the minimization of the torque ripple. To do so, the time point at which the switches of the inverter should change state is calculated. The proposed control scheme, called variable switching point predictive torque control (VSP^{2}TC), is employed to control both a two-level inverter driving an induction machine

(IM), as well as a three-level neutral point clamped (NPC) voltage source inverter driving an IM.

Περίληψη

Αυτή η εργασία εστιάζει σε στρατηγικές προβλεπτικού ελέγχου (model predictive control—MPC) που εφαρμόζονται σε μετατροπείς ηλεκτρονικών ισχύος και στην οδήγηση ενός κινητήρα επαγωγής. Λόγω της διακοπτικής τους φύσης, τα συστήματα ηλεκτρονικών ισχύος αποτελούν μη γραμμικά συστήματα με πολλαπλές εισόδους και εξόδους, ενώ είναι και υποκείμενο σε διάφορους περιορισμούς (π.χ. ο βαθμός χρησιμοποίησης ενός διακόπτη πρέπει να είναι μεταξύ του μηδενός και του ένα). Κατά την πάροδο των χρόνων αναπτύχθηκαν πολλές αποτελεσματικές τεχνικές ελέγχου, οι οποίες βασίζονται κυρίως σε γραμμικούς ελεγκτές, όπως ο αναλογικός-ολοκληρωτικός-διαφορικός (PID), σε συνδυασμό με μη γραμμικές τεχνικές, όπως η διαμόρφωση εύρους παλμών (pulse width modulation—PWM). Παρ' όλα αυτά, οι ελεγκτές αυτού του τύπου επιτυγχάνουν βέλτιστη απόδοση μόνο σε ένα μικρό εύρος λειτουργίας, καθώς εκτός αυτού του εύρους λειτουργίας η απόδοση μειώνεται αισθητά. Επομένως, τα προβλήματα που σχετίζονται με πολλές εφαρμογές και τη συμπεριφορά τους σε κλειστό βρόχο ακόμα αποτελούν προκλήσεις τόσο σε θεωρητικό, όσο και σε πρακτικό επίπεδο.

Μία τεχνική ελέγχου η οποία εφαρμόζεται ευρέως τα τελευταία χρόνια στο πεδίο των ηλεκτρονικών ισχύος είναι ο προβλεπτικός έλεγχος. Ο προβλεπτικός έλεγχος είναι μία στρατηγική ελέγχου η οποία αναπτύχθηκε ως μία εναλλακτική πρόταση στις συμβατικές τεχνικές βασιζόμενες στους PID ελεγκτές. Η επιτυχία του βασίζεται στο ότι χρησιμοποιεί το μαθηματικό μοντέλο του συστήματος, με αποτέλεσμα να προβλέπεται με επιτυχία η επίδραση της μεταβλητής εισόδου. Επιπλέον, ο προβλεπτικός έλεγχος μπορεί και χειρίζεται σύνθετες και μη γραμμικές δυναμικές, ενώ ο σαφής ορισμός σχεδιαστικών κριτηρίων και περιορισμών γίνεται με σχετικά απλό και αποτελεσματικό τρόπο. Επιβάλλοντας περιορισμούς στις μεταβλητές που ενδιαφέρουν το σχεδιαστή, το σύστημα μπορεί να λειτουργεί στα φυσικά του όρια, χωρίς όμως να τα παραβιάζει. Επομένως, η καλύτερη δυνατή λειτουργία επιτυγχάνεται, ενώ λαμβάνονται υπ' όψιν τα όρια λειτουργίας του συστήματος. Λόγω των προαναφερθέντων πλεονεκτημάτων η συγκεκριμένη τεχνική ελέγχου έχει τραβήξει το

ενδιαφέρον της επιστημονικής και ερευνητικής κοινότητας. Επιπροσθέτως, η ανάπτυξη ταχύτερων μικροεπεξεργαστών έχει σαν αποτέλεσμα την επιτυχή υλοποίηση του εν λόγω υπολογιστικά απαιτητικού αλγορίθμου.

Η παρούσα διατριβή αποτελείται από δύο μέρη. Στο πρώτο, παρουσιάζονται οι βασικές έννοιες του προβλεπτικού ελέγχου, συμπεριλαμβανομένων των εννοιών της βελτιστοποίησης, του βέλτιστου ελέγχου και του μετατιθέμενου ορίζοντα. Επιπλέον, εμπεριέχεται μία σύντομη εισαγωγή στη μοντελοποίηση υβριδικών συστημάτων ως υβριδικά αυτόματα, καθώς και στη στρατηγική απαρίθμησης των πιθανών καταστάσεων.

Το δεύτερο μέρος αφιερώνεται σε εφαρμογές του προβλεπτικού ελέγχου στο πεδίο των ηλεκτρονικών ισχύος. Αποτελείται από τρία κεφάλαια, κάθε ένα από τα οποία αναφέρεται σε διαφορετική εφαρμογή. Συγκεκριμένα, το Κεφάλαιο 3 αναφέρεται στους μετατροπείς συνεχούς τάσης σε συνεχή, οι οποίοι με κατάλληλο έλεγχο επιτυγχάνουν ανύψωση της τάσης εξόδου (dc-dc boost converters), το Κεφάλαιο 4 στους πολυεπίπεδους ανορθωτές με αλληλένδετες βαθμίδες (cascaded H-bridge multilevel rectifiers), ενώ το Κεφάλαιο 5 στην οδήγηση κινητήρων επαγωγής (ac drives).

Στο Κεφάλαιο 3 παρουσιάζονται δύο αλγόριθμοι προβλεπτικού ελέγχου για τους μετατροπείς συνεχούς τάσης σε συνεχή. Ο μετατροπέας μοντελοποιείται σαν ένα υβριδικό αυτόματο διακριτού χρόνου. Το συγκεκριμένο μαθηματικό μοντέλο περιγράφει με ακρίβεια τη λειτουργία του μετατροπέα τόσο σε κατάσταση συνεχούς αγωγής (continuous conduction mode—CCM), όσο και σε κατάσταση ασυνεχούς αγωγής (discontinuous conduction mode—DCM). Σχεδιάζεται μία αντικειμενική συνάρτηση και ελαχιστοποιείται όντας υποκείμενο στις δυναμικές του μοντέλου. Σύμφωνα με τον πρώτο προτεινόμενο αλγόριθμο, ο ελεγκτής υλοποιείται ως ελεγκτής ρεύματος. Δύο βρόχοι σχεδιάζονται, με τον εσωτερικό να βασίζεται στον προβλεπτικό έλεγχο. Δύο διαφορετικές αντικειμενικές συναρτήσεις προτείνονται και μελετώνται. Ο στόχος του ελέγχου, που είναι η ρύθμιση του ρεύματος στην αναφορά του, επιτυγχάνεται με άμεσο έλεγχο του διακόπτη, επομένως δε χρειάζεται ένας διαμορφωτής παλμών. Στη δεύτερη προτεινόμενη στρατηγική, που υλοποιείται ως ελεγκτής τάσης, η ρύθμιση της τάσης στην αναφορά της επιτυγχάνεται άμεσα, χωρίς τη χρήση βρόχου ρεύματος. Επιπλέον, και για τις δύο μεθοδολογίες, υλοποιείται ένας εκτιμητής για την αποφυγή προβλημάτων λόγω αβεβαιοτήτων που οφείλονται στο φορτίο και αστοχιών του μοντέλου.

Στο Κεφάλαιο 4 σχεδιάζεται μία στρατηγική προβλεπτικού ελέγχου για τον πολυεπίπεδο ανορθωτή με αλληλένδετες βαθμίδες. Ο προτεινόμενος αλγόριθμος φροντίζει ώστε το ημιτονοειδές ρεύμα εισόδου να παραμένει σε φάση με την τάση εισόδου, ενώ ταυτόχρονα επιτυγχάνεται ανεξάρτητη ρύθμιση των τάσεων των κελιών του μετατροπεά στις αναφορές τους, μέσω άμεσου ελέγχου των διακοπτών, χωρίς την παρουσία διαμορφωτή παλμών. Επιπλέον, επειδή όλοι οι διακοπτικοί συνδυασμοί λαμβάνονται υπ' όψιν ο ελεγκτής έχει την επιθυμητή συμπεριφορά όχι μόνο υπό ονομαστικές συνθήκες, αλλά και υπό ασύμμετρες

τάσεις και ανόμοια φορτία. Τέλος, για την εξασφάλιση της ευρωστίας του ελεγκτή χρησιμοποιείται ένας μικρός ορίζοντας πρόβλεψης. Με αυτόν τον τρόπο το υπολογιστικό κόστος παραμένει σε λογικά πλαίσια, με αποτέλεσμα να είναι εφικτή η εκτέλεση του αλγορίθμου σε πραγματικό χρόνο.

Στο Κεφάλαιο 5 παρουσιάζεται μία μέθοδος που εφαρμόζεται στον προβλεπτικό έλεγχο ροπής (predictive torque control—PTC), σύμφωνα με την οποία υπολογίζεται ένα χρονικά μεταβαλλόμενο σημείο στο οποίο οι διακόπτες του μετατροπέα αλλάζουν κατάσταση. Στον προβλεπτικό έλεγχο ροπής η διακοπτική συχνότητα περιορίζεται από τη συχνότητα δειγματοληψίας, καθώς η θεωρητικά μέγιστη τιμή της είναι ίση με τη μισή τιμή της συχνότητας δειγματοληψίας. Εν τούτοις, στην πραγματικότητα η διακοπτική συχνότητα είναι μικρότερη από αυτήν την τιμή, με αποτέλεσμα να προκύπτουν υψηλές κυματώσεις στο ρεύμα και στη ροπή εν συγκρίσει με τις μεθόδους όπου χρησιμοποιείται διαμορφωτής εύρους παλμών. Για να αποφευχθεί αυτό το πρόβλημα προτείνεται ένα πρόβλημα βελτιστοποίησης, το οποίο λύνεται σε πραγματικό χρόνο. Στόχος του συγκεκριμένου προβλήματος βελτιστοποίησης είναι εκτός από τη ρύθμιση της ροπής και της ροής του στάτη στις αναφορές τους, η ελαχιστοποίηση της κυμάτωσης της ροπής. Επομένως, με την εν λόγω μεθοδολογία υπολογίζεται το βέλτιστο χρονικό σημείο, εντός μίας περιόδου δειγματοληψίας, κατά το οποίο οι διακόπτες του αντιστροφέα πρέπει να αλλάξουν κατάσταση ώστε να ικανοποιηθούν και οι τρεις προαναφερθέντες στόχοι. Ο προτεινόμενος αλγόριθμος, που ονομάζεται προβλεπτικός έλεγχος ροπής μεταβλητού διακοπτικού σημείου (variable switching point predictive torque control—VSP^{2}TC), εφαρμόζεται σε ένα σύστημα αντιστροφέα δύο επιπέδων-κινητήρα επαγωγής, καθώς και σε ένα σύστημα αντιστροφέα τριών επιπέδων με διόδους περιορισμού-κινητήρα επαγωγής.

Contents

Chapter 1

Introduction

Power electronics is a mature technology that has been in use for more than four decades. From air-conditioners to rail transport and from mobile phones to motor drives, power electronics circuits have proved indispensable in many areas because they convert electrical power from one form to another, such as ac-dc, dc-dc, dc-ac, or even ac-ac with a variable output magnitude and frequency [103].

Over the years many control strategies for power electronics have been proposed that have been shown to be reasonably effective. Mainly, these are strategies based on linear controllers combined with nonlinear techniques, such as pulse width modulation (PWM). However, controllers of this type are usually tuned to achieve optimal performance only over a narrow operating range; outside this range the performance is significantly deteriorated. Therefore, the problems associated with many applications and their closed-loop controlled performance still poses theoretical and practical challenges. Furthermore, the advent of new applications leads to the need for new control approaches that will meet the increasingly demanding performance requirements.

A control algorithm that has been recently gaining more popularity in the field of power electronics is model predictive control (MPC) [95, 118]. This control method, which has been successfully used in the process industry since the 1970s, has attracted the interest and attention of research and academic communities due to its numerous advantageous features, such as design simplicity, explicit inclusion of design criteria and restrictions, fast dynamics and inherent robustness. In addition, the emergence of fast microprocessors has increasingly enabled successful implementation [30, 40, 77, 92].

In MPC, an optimization problem is formulated based on an objective function that captures the control objectives over a finite prediction horizon. The control action is determined by minimizing in real-time and at every time-step the chosen objective function, subject to the discrete-time model of the system and constraints. The sequence of control inputs with the minimum associated cost is the *optimal* solution. Out of this

sequence only the first element is applied to the converter. In the next sampling instant, all the variables are shifted by one sampling interval and the optimization problem is repeated based on new measurements or estimates. This procedure is known as the *receding horizon policy* [99]. In this way feedback is provided, allowing one to cope with model uncertainties and disturbances.

This thesis focuses on MPC strategies in the field of power electronics. More specifically, MPC-based algorithms are employed for controlling dc-dc boost converters, single phase cascaded H-bridge (CHB) multilevel rectifiers, and ac low voltage (LV) drives.

1.1 DC-DC Converters

Introduction

The control of dc-dc converters constitutes a challenging task, due to their switched nonlinear (or hybrid) characteristic. The main control objective is the regulation of the output voltage to a commanded value, while rejecting the impact of variations in the input voltage and the load. This can be achieved either by directly controlling the voltage, i.e. by employing a single loop, or by indirectly controlling the voltage by controlling the inductor current, i.e. two loops are required; the outer—voltage—loop adjusts the current reference for the inner loop such that the output voltage is regulated to its desired reference. The inner—current—loop drives the inductor current to its reference, by manipulating the switch.

In both cases the standard control approach is to average the continuous-time dynamics associated with the different modes of operation, and to linearize them about the operating point. A different approach is to directly address the hybrid nature of these converters, see e.g. [46]. Thus, thanks to these recent theoretical advances in the control of hybrid systems, as well as to the emergence of fast microprocessors, the application of MPC to the field of dc-dc converters has been enabled.

Contributions

In this thesis, MPC is employed both as a voltage-mode controller, as well as a current-mode controller, as presented in Chapter 3. The discrete-time switched (hybrid) mathematical model of the converter, which serves as a prediction model for MPC, captures all operating modes of the inductor current, making it suitable for operation both in the continuous (CCM) and discontinuous conduction mode (DCM). Hence, the converter state can be accurately predicted for the whole operating regime. For both MPC schemes the converter switch is directly manipulated in order to meet the control objective.

In both strategies the most prominent drawback is the inherent computational complexity since the computational power required increases exponentially as the prediction

horizon is extended. With regards to the voltage-mode MPC scheme, a move blocking strategy is adopted [20] to address this issue, which results in a significant reduction of the computations required and facilitates the real-time implementation of the controller. On the other hand, for the current-mode MPC scheme, a relatively small prediction horizon suffices in order to obtain a satisfactory control result. Moreover, the controllers are augmented by a load estimation scheme, namely a discrete-time switched Kalman filter, suitable for all operating modes. The Kalman filter is added to estimate the converter states and to provide offset-free tracking of the output voltage due to its integrating action, despite changes in the load. In that way the robustness of the controller is ensured even when the converter operates under nonnominal conditions.

Publications

Chapter 3 is mainly based on [63], [64], and [66]. Alternative MPC-based strategies applied to several dc-dc converter topologies, such as the interleaved dc-dc boost converter with uncoupled and coupled inductors, can be found in [67], [68], and [65].

1.2 Cascaded H-Bridge Multilevel Rectifiers

Introduction

Multilevel converters have turned into a mature technology designed to be used in applications where a high power demand is required. Thanks to multilevel converters it is possible to work at voltage levels beyond the classic semiconductor limits, since they synthesize a staircase voltage waveform, with each voltage step lying within the ratings of the individual power devices. Three are the major multilevel converter topologies: the neutral point clamped (NPC) [106], the flying capacitor (FC) [101], and the cascaded H-bridge (CHB) [52] converters. The CHB topology is particularly attractive due to its high modularity, simplicity, and the minimum number of components required to synthesize the same number of voltage levels with the other multilevel topologies. Moreover, when operated as a rectifier it can be applied to a wide range of applications such as industrial plants, traction systems, and uninterruptible power supplies.

However, control of CHB converters, especially when operated as rectifiers, still poses several challenges, due to the inherent coupling of the cascade-connected modules (or cells) and the highly nonlinear dynamics. A proper controller should aim to achieve independent operation of the n isolated dc buses. This means that the controller should be able to achieve accurate operation, i.e. accurate regulation of the cell output voltages to their demanded values, even under asymmetrical voltage potentials and unbalanced loads. Furthermore, the input current should be sinusoidal and in phase with the supply voltage over the entire operating regime, i.e. operation with unity power factor should be

maintained, and its harmonic content should be kept as low as possible.

Contributions

To meet the aforementioned goals, in Chapter 4 an MPC scheme for the CHB multilevel rectifier consisting of n cells is proposed. A detailed discrete-time state-space model suitable for the controller is derived. In the inner loop, posed in the MPC framework, the input current is regulated to its sinusoidal reference—derived from the proportional-integral (PI) based outer, voltage loop—by directly manipulating the switches of the converter without the need of a modulator. Furthermore, since all the possible switching combinations are enumerated and taken into account, the controller exhibits favorable performance not only under nominal conditions, but also under asymmetrical voltage potentials and unbalanced loads. In order to further improve the dynamic performance of the plant a voltage term is added in the formulated objective function; thanks to this term the controller aims to eliminate the nonzero voltage error by fast charging the capacitors, and it ensures a zero steady-state voltage tracking error.

In order to cope with the computational complexity inherent in the proposed MPC formulation a short horizon is employed. The length of the horizon suffices to ensure robustness, while the control result does not deteriorate. In this way the required computational effort remains reasonable, making it possible to implement the algorithm in a real-time system. In addition, solutions to significantly reduce the number of feasible switching transitions, i.e. the transitions to be evaluated in real-time, such as imposing *a priori* constraints to them, are proposed, and their effect on the dynamic behavior of the system is highlighted.

Publications

Chapter 4 is mostly based on [69].

1.3 AC Drives

Introduction

Adjustable-speed ac drives based on voltage source inverters are in constantly increasing demand for numerous industrial applications. Thereby, control schemes that provide robustness and favorable dynamic behavior, among others, are required. Two of the most common control algorithms are field oriented control (FOC) [73] and direct torque control (DTC) [127]. The first is a modulator-based control approach showing better performance with respect to current, torque and flux ripples. The latter is a direct control strategy where the state of the switches is selected from look-up tables, exhibiting great behavior during transients.

During the last decade many MPC approaches have been presented to tackle issues such as design simplicity and explicit inclusion of constraints that stem from the drive. A well-known approach is the so-called predictive torque control (PTC) [29, 102], which is a strategy based on DTC, but the design process is more straightforward and simpler. The main control objectives are the same, i.e. the regulation of the torque and stator flux magnitude to their reference values, and captured in an objective function. Thus, the use of hysteresis bounds and look-up tables is omitted. Nonetheless, since PTC is a direct switching strategy, the delivered torque ripples are still high, a fact that is of concern when LV ac drive systems are considered. A simple solution to overcome this obstacle is to increase the switching frequency, since switching losses, that are directly related to the switching frequency, are of less importance in LV drives. This means, however, that the sampling interval should be further decreased; an option which would lead to a more costly hardware.

Contributions

In this thesis an MPC-based algorithm, named variable switching point predictive torque control (VSP^{2}TC), is presented in Chapter 5 that aims not only to fulfill the main objectives mentioned above, but also to minimize the torque ripple. This is achieved by allowing the switchings to take place *in between* the sampling interval. To do so, an objective function that encompasses all the variables of concern is formulated and minimized in real-time. Based on the aforementioned optimization problem, a time instant that lies between the discrete time instants is calculated; at this instant the switches should change state. In this way the ripple can be decreased without the need of a further reduction of the sampling interval. It should be mentioned, though, that the introduced scheme comes with a slightly increased switching frequency compared to PTC. Nonetheless, this consequent increase in the switching frequency is limited by the sampling frequency, since its maximum value is equal to the half of it.

Finally, in Chapter 5, two examples are included to highlight the performance of the proposed strategy: a two-level inverter driving an induction motor (IM) and a three-level NPC inverter driving an IM. Note that for the three-level NPC inverter-IM system an additional control objective should be considered, which is the balancing of the neutral point potential. Therefore, the initial algorithm introduced for the two-level inverter is refined accordingly to meet all the control objectives.

Publications

Chapter 5 is largely based on [70] and [125].

Part I

Theoretical Background

Chapter 2

Theoretical Background

In this chapter a brief theoretical background is provided; the basic concepts and terminology related to model predictive control (MPC) are presented.

2.1 Basic Definitions

Firstly, some basic definitions are presented based on [17].

Definition 2.1. (Line): Suppose $\boldsymbol{x}_1 \neq \boldsymbol{x}_2$ are two vectors in $\mathbb{R}^n$. The *line* through them is the set of the form

$$\{\boldsymbol{x} \mid \boldsymbol{x} = \theta \boldsymbol{x}_1 + (1-\theta)\boldsymbol{x}_2,\ \theta \in \mathbb{R}\}\,.$$

Definition 2.2. (Line Segment): Let $\boldsymbol{x}_1 \neq \boldsymbol{x}_2 \in \mathbb{R}^n$, and $\theta \in \mathbb{R}$. For values of the parameter θ between 0 and 1, the *line segment* through $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is a set defined as

$$\{\boldsymbol{x} \mid \boldsymbol{x} = \theta \boldsymbol{x}_1 + (1-\theta)\boldsymbol{x}_2,\ 0 \leq \theta \leq 1\}\,.$$

Definition 2.3. (Hyperplane): A *hyperplane* in $\mathbb{R}^n$ is defined as the set

$$\{\boldsymbol{x} \mid \boldsymbol{a}^T \boldsymbol{x} = b\}\,,$$

where $\boldsymbol{a} \in \mathbb{R}^n,\ \boldsymbol{a} \neq \mathbb{0}$ ($\mathbb{0}$ is the zero vector), and $b \in \mathbb{R}$.

Definition 2.4. (Convex Set): A set $\mathcal{C}$ is *convex* if the line segment joining any two points in $\mathcal{C}$ lies in $\mathcal{C}$, i.e. if for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{C}$ and any θ with $0 \leq \theta \leq 1$, we have

$$\theta \boldsymbol{x}_1 + (1-\theta)\boldsymbol{x}_2 \in \mathcal{C}\,.$$

Definition 2.5. (Convex Function): A function $f : \mathbb{R}^n \to \mathbb{R}$ is *convex* if $\mathbf{dom}\, f$ is a convex set and if for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbf{dom}\, f$, and θ with $0 \leq \theta \leq 1$, we have

$$f(\theta \boldsymbol{x} + (1-\theta)\boldsymbol{y}) \leq \theta f(\boldsymbol{x}) + (1-\theta) f(\boldsymbol{y})\,. \tag{2.1}$$

A function f is *strictly convex* if strict inequality holds in (2.1) whenever $\boldsymbol{x} \neq \boldsymbol{y}$ and $0 < \theta < 1$.

Definition 2.6. (Affine Function): A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *affine* if it is a sum of a linear function and a constant, i.e. if it has the form

$$f(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}\,,$$

where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{b} \in \mathbb{R}^m$.

2.2 Hybrid Systems and Modeling

In a large number of application areas engineers are dealing with systems that involve the interaction of different types of dynamics. These systems are constituted not only by parts governed by difference or differential equations, but also by parts described by logic, such as on/off switches, finite state machines, if-then-else rules, and yes/no conditions. These systems, called *hybrid* systems, are hierarchical systems that feature heterogeneous set of dynamics; the lower level includes the continuous-valued components, while the upper level the discrete-valued components. Hybrid systems switch between different modes of operation, where each mode is governed by its own characteristic dynamical law. In order to switch from one mode to another certain conditions should be met; in general, mode transitions are triggered by variables crossing specific thresholds (state events), by the elapse of certain time periods (time events), and/or by external inputs (input events) [49, 90].

For purposes of hybrid systems modeling several approaches have been proposed. The resulting models can be linear or nonlinear, depending on the dynamics of the system and the constraints [54]. However, by considering the dynamics of the model to be piecewise affine, rather nonlinear, numerically tractable schemes can be designed, whereas problems, such as Zeno behavior[1], can be avoided.

Hybrid systems that can be described by linear dynamic equations subject to mixed-integer inequalities, i.e. inequalities that include both continuous and binary variables, can be modeled as mixed logical dynamical (MLD) systems [12]. Systems that can

[1] Zeno behavior occurs if an infinite number of discrete transitions (or jumps) occur in a finite amount of time, i.e. if the switching times converge. For more details on Zeno behavior the reader is referred to [4, 83, 90].

be described by partitioning the state-space into polyhedra, each of which is described by its own affine functions, can be modeled as polyhedral piecewise affine [123]. Systems where the switching between different dynamics is governed by a finite automaton can be modeled as hybrid automata [2]. Other types of modeling include linear complementary (LC) systems, max-min-plus-scaling (MMPS) systems, and extended linear complementary (ELC) systems. It should be mentioned, though, that the foregoing modeling approaches are equivalent—in some cases by imposing some mild assumptions—as it is shown in [10, 11, 130]. Thereby, models such as linear hybrid systems, finite state machines, and nonlinear systems that can be expressed as piecewise linear (or affine) functions can be modeled based on one of the approaches mentioned before. Finally, in [82] a number of formalisms that have been developed for modeling hybrid systems are presented.

In this work the hybrid systems under investigation are modeled as hybrid automata. Thus, in the following, the characteristics of an automaton are briefly presented.

2.2.1 Hybrid Automata

Models the state of which can be explicitly partitioned into a continuous state $\boldsymbol{x}$ and a discrete state $\boldsymbol{q}$ are called *hybrid automata*. Since the nature of these systems is a combination of continuous time systems and discrete event systems, tools from the fields of control theory and computer science can be used for their modeling and analysis.

For each discrete state $\boldsymbol{q} \in \mathcal{Q}$, where $\mathcal{Q} = \{q_1, \ldots, q_N\}$ is a finite set of the discrete states, the values of the continuous state $\boldsymbol{x} \in \mathcal{X}$, with $\mathcal{X} \subseteq \mathbb{R}^n$ being the continuous state-space, are specified. Based on the above, the differential equations that describe the dynamics of the model in the continuous-time domain and depend on the discrete state $\boldsymbol{q}$ are of the form

$$\frac{d\boldsymbol{x}}{dt} = f_q\big(\boldsymbol{x}(t)\big) = \boldsymbol{A}_q\boldsymbol{x}(t) + \boldsymbol{b}_q\,, \tag{2.2}$$

where $\boldsymbol{A}_q \in \mathbb{R}^{n\times n}$, and $\boldsymbol{b}_q \in \mathbb{R}^n$. Equivalently, in the discrete-time domain the system is described by the difference equation

$$\boldsymbol{x}(k+1) = f_{q_d}\big(\boldsymbol{x}(k)\big) = \boldsymbol{A}_{q_d}\boldsymbol{x}(k) + \boldsymbol{b}_{q_d}\,, \tag{2.3}$$

where $\boldsymbol{A}_{q_d} \in \mathbb{R}^{n\times n}$ is derived from the $\boldsymbol{A}_q$ matrix, and $\boldsymbol{b}_{q_d} \in \mathbb{R}^n$ from $\boldsymbol{b}_q$. It should be noted that the dynamics of the state-update function f_q (or f_{q_d}) can be applied to a subset of the continuous state space $\mathcal{X}$, i.e. $\mathcal{J}(\boldsymbol{q}) \subseteq \mathcal{X}$.

In order to explicitly define a hybrid automaton the following data are required [53,94]:

- A set of *discrete states* $\mathcal{Q}$.
- A set of *continuous states* $\mathcal{X}$.

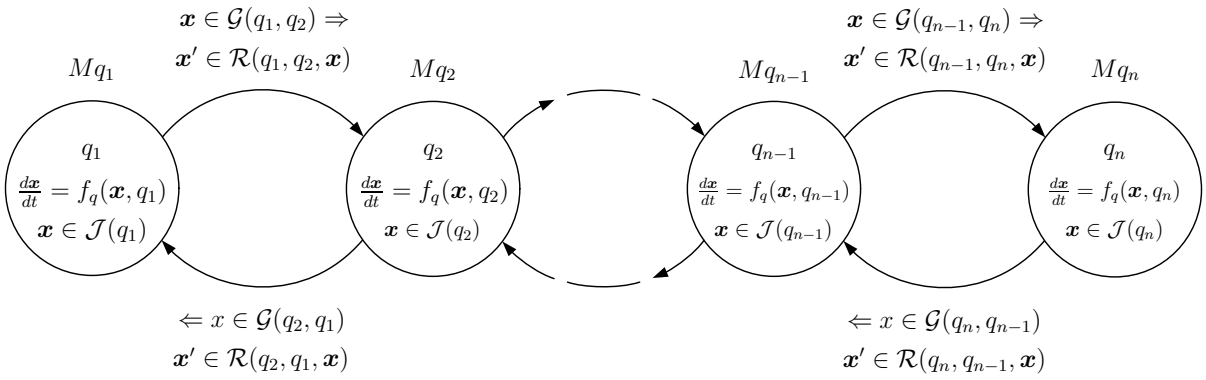


Figure 2.1: A hybrid automaton with n modes of operation $M_{q_1}, \ldots, M_{q_n}$. In each mode the state $\boldsymbol{x}$ evolves according to the corresponding state-update function in the set domain, i.e. $\frac{d\boldsymbol{x}}{dt} = f_q(\boldsymbol{x}, q_i)\ \forall\ \boldsymbol{x} \in \mathcal{J}(g_i)$, with $i = 1, \ldots, n$. For the transition from mode M_{q_i} to mode M_{q_j}, with $j \in \{i-1, i+1\}$, the condition $\boldsymbol{x} \in \mathcal{G}(q_i, q_j)$ must be fulfilled, while the state changes: $\boldsymbol{x}' \in \mathcal{R}(q_i, q_j, \boldsymbol{x})$.

- A set of *modes* M_q, with $\boldsymbol{q} \in \mathcal{Q}$, where a mode is the operation of the system given by (2.2) (or (2.3)) for every $\boldsymbol{x} \in \mathcal{J}(\boldsymbol{q})$.

- A *vector field* $f : \mathcal{Q} \times \mathcal{X} \to \mathbb{R}^n$, which describes through the differential equation (2.2) (or the difference equation (2.3)) the evolution of the continuous state $\boldsymbol{x}$.

- A set of *initial states* $\mathcal{I} \subseteq \mathcal{Q} \times \mathcal{X}$.

- A *domain map* $\mathcal{J} : \mathcal{Q} \to P(\mathcal{X})$, where $P(\mathcal{X})$ is the power set of $\mathcal{X}$, i.e. the set of all subsets of $\mathcal{X}$.

- A set of *edges* $\mathcal{E} \subseteq \mathcal{Q} \times \mathcal{Q}$, which is the collection of the feasible transitions from mode M_q to mode $M_{q'}$.

- A *guard map* $\mathcal{G} : \mathcal{E} \to P(\mathcal{X})$, which assigns to each edge $e = (q, q') \in \mathcal{E}$ a guard, i.e. it identifies the set $\mathcal{G}(q, q')$ to which the continuous state $\boldsymbol{x}$ should belong so that a transition from mode M_q to mode $M_{q'}$ is feasible.

- A *reset map* $\mathcal{R} : \mathcal{E} \times \mathcal{X} \to P(\mathcal{X})$, which gives the value assigned to the continuous state $\boldsymbol{x} \in \mathbb{R}^n$ during a transition from mode M_q to mode $M_{q'}$, for each edge $e = (q, q') \in \mathcal{E}$.

Based on the above a hybrid automaton can be defined as:

Definition 2.7. (Hybrid Automaton): A *hybrid automaton* $\mathscr{H}$ is a collection $\mathscr{H} = (\mathcal{Q}, \mathcal{X}, f, \mathcal{I}, \mathcal{J}, \mathcal{E}, \mathcal{G}, \mathcal{R})$.

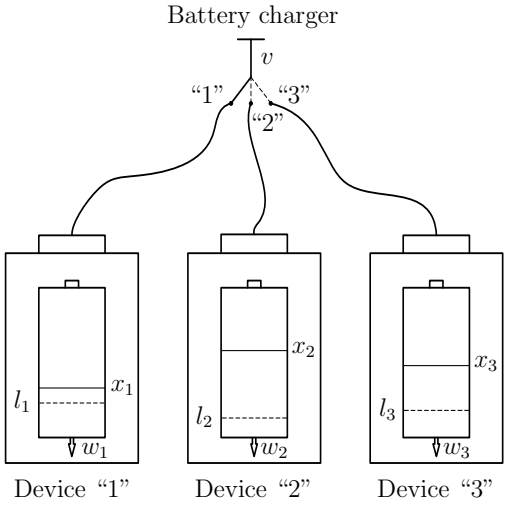


Figure 2.2: Battery charger system.

In Fig. 2.1 a system modeled as a hybrid automaton that consists of n modes of operation is depicted. As can be seen, the continuous dynamics of the n operating modes, the transition conditions that need to be met in order to move from one operating mode to the other, and the reset rules that govern transitions between these modes are shown.

Example 2.1. (Battery Charger): Three electronic devices use rechargeable batteries as a power source. The batteries of all three devices are connected to the same battery charger. However, only one device at a time can be connected to the charger; an automatically operated switch is used to connect the one device, and disconnect the others, and it is assumed that this can happen instantaneously. Thus, at any point in time the battery of only one device is charging at a constant rate through the charger. Furthermore, the batteries of the devices are discharging at a constant rate. The whole system is shown in Fig. 2.2.

Let x_i, with $i = \{1, 2, 3\}$, denote the charge of the ith battery, $w_i > 0$ the constant flow of charge out of the ith battery, and v the constant flow of charge into the system. Initially the charge of the three batteries is higher than a lower limit (lower state of charge limit), which is l_1 for the first battery, l_2 for the second, and l_3 for the third. The objective is to not allow the charge of the batteries to get below their lower limits. Thereby, a controller is employed that connects the first battery to the charger whenever $x_1 \leq l_1$, the second whenever $x_2 \leq l_2$, and the third whenever $x_3 \leq l_3$.

Based on definition 2.7 the hybrid automaton that describes the process is defined as follows:

- Discrete states: $\mathcal{Q} = \{q_1, q_2, q_3\}$; the switch is at position "1" (the charger is connected to the first device), or at position "2" (the charger is connected to the second device), or at position "3" (the charger is connected to the third device).

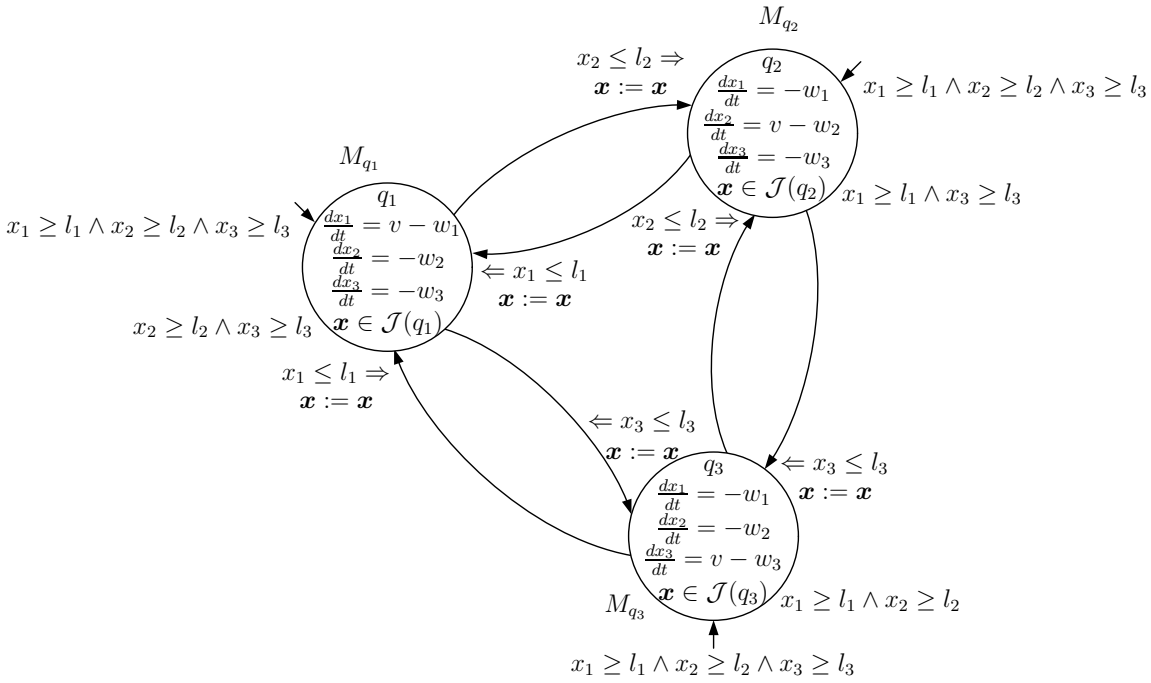


Figure 2.3: The battery charger system represented as a hybrid automaton.

- Continuous states: $\mathcal{X} = \mathbb{R}^3$; the state of charge of the batteries.

- Modes of operation: M_{q_1}, M_{q_2} & M_{q_3}; one mode for the case where the charger is connected to the first device, one mode when the charger is connected to the second one, and one mode when it is connected to the third one.

- Vector field: $f(q_1, \boldsymbol{x}) = \begin{bmatrix} v - w_1 \\ -w_2 \\ -w_3 \end{bmatrix}$, $f(q_2, \boldsymbol{x}) = \begin{bmatrix} -w_1 \\ v - w_2 \\ -w_3 \end{bmatrix}$, and $f(q_3, \boldsymbol{x}) = \begin{bmatrix} -w_1 \\ -w_2 \\ v - w_3 \end{bmatrix}$; when the switch is at position "1" the charge of the first battery increases, while the charge of the second and the third battery decreases, and so on.

- Initial states: $\mathcal{I} = \{q_1, q_2, q_3\} \times \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_1 \geq l_1 \wedge x_2 \geq l_2 \wedge x_3 \geq l_3\}$; at $t = 0$ the charge of all three batteries is above the lower limits l_1, l_2, and l_3.

- Domain map: $\mathcal{J}(q_1) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_2 \geq l_2 \wedge x_3 \geq l_3\}$, $\mathcal{J}(q_2) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_1 \geq l_1 \wedge x_3 \geq l_3\}$, and $\mathcal{J}(q_3) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_1 \geq l_1 \wedge x_2 \geq l_2\}$; one battery is charged, as long as the charge of the other two batteries is above their lower limits.

- Edges: $\mathcal{E} = \{(q_1, q_2), (q_1, q_3), (q_2, q_1), (q_2, q_3), (q_3, q_1), (q_3, q_2)\}$: the switch can change from position "1" to position "2" or to position "3", or from position "2" to position "1" or to position "3", and so on.

- Guard map: $\mathcal{G}(q_1, q_2) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_2 \leq l_2\}$, $\mathcal{G}(q_1, q_3) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_3 \leq l_3\}$, $\mathcal{G}(q_2, q_1) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_1 \leq l_1\}$, $\mathcal{G}(q_2, q_3) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_3 \leq l_3\}$, $\mathcal{G}(q_3, q_1) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_1 \leq l_1\}$, and $\mathcal{G}(q_3, q_2) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid x_2 \leq l_2\}$; when the state of charge of one battery reaches its lower limit, the charger is connected to it.

- Reset map: $\mathcal{R}(q_1, q_2, \boldsymbol{x}) = \mathcal{R}(q_1, q_3, \boldsymbol{x}) = \mathcal{R}(q_2, q_1, \boldsymbol{x}) = \mathcal{R}(q_2, q_3, \boldsymbol{x}) = \mathcal{R}(q_3, q_1, \boldsymbol{x}) = \mathcal{R}(q_3, q_2, \boldsymbol{x}) = \{\boldsymbol{x}\}$; the continuous state remains unaffected due to the switching charging.

The battery charger system represented as hybrid automaton is shown in Fig. 2.3. ■

2.3 Optimization Problems

Model predictive control is an optimal control scheme. Therefore, before introducing the concepts of model predictive control, it is necessary to recall the basic terminology for mathematical optimization theory. Furthermore, some important classes of optimization problems, namely convex optimization problems, linear optimization problems, quadratic optimization problems, mixed-integer linear optimization problems, and mixed-integer quadratic optimization problems are presented. For further details on optimization theory, applications and algorithms the reader is referred to [13–15, 17, 38, 93].

2.3.1 Basic Terminology

According to [17], an optimization problem is of the form

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{x}) & \\ \text{subject to} & g_i(\boldsymbol{x}) \leq 0, & i = 1, \ldots, m \\ & h_j(\boldsymbol{x}) = 0, & j = 1, \ldots, p\,. \end{array} \tag{2.4}$$

The goal is to find the *optimization variable* $\boldsymbol{x} \in \mathbb{R}^n$ that minimizes the *objective* (or *cost*) function $f : \mathbb{R}^n \to \mathbb{R}$, while satisfying the conditions $g_i(\boldsymbol{x}) \leq 0,\ i = 1, \ldots, m$, and $h_j(\boldsymbol{x}) = 0,\ j = 1, \ldots, p$. The inequalities $g_i(\boldsymbol{x}) \leq 0$ in (2.4) are called *inequality constraints*, and the corresponding functions $g_i : \mathbb{R}^n \to \mathbb{R}$ *inequality constraints functions*, while the equalities $h_j(\boldsymbol{x}) = 0$ are called *equality constraints*, and the corresponding functions $h_j : \mathbb{R}^n \to \mathbb{R}$ *equality constraints functions*. Finally, for the case where $m = p = 0$, i.e. there are no constraints, the optimization problem (2.4) is called *unconstrained*.

The *domain* $\mathcal{O}$ of the optimization problem (2.4) is the set of the points for which the objective function f and the constraint functions g and h are defined, thus

$$\mathcal{O} = \mathbf{dom}\, f \cap \bigcap_{i=1}^{m} \mathbf{dom}\, g_i \cap \bigcap_{j=1}^{p} \mathbf{dom}\, h_j\,.$$

A point $\boldsymbol{x} \in \mathcal{O}$ is said to be *feasible* if it satisfies all the constraints $g_i(\boldsymbol{x}) \leq 0,\ i = 1, \ldots, m$, and $h_j(\boldsymbol{x}) = 0,\ j = 1, \ldots, p$. The problem (2.4) is *feasible* if there exists at least one feasible point, else it is *infeasible*. The set of all feasible points is called the *feasible set*.

The *optimal value* q^* of the problem (2.4) is defined as

$$q^* = \inf\{f(\boldsymbol{x}) \mid g_i(\boldsymbol{x}) \leq 0,\ i = 1, \ldots, m,\ h_j(\boldsymbol{x}) = 0,\ j = 1, \ldots, p\}\,.$$

The optimal value q^* may be equal to $\pm\infty$. If the problem is infeasible then $q^* = \infty$; if the problem is *unbound below*, i.e. there are points $\boldsymbol{x}_k$ such that $f(\boldsymbol{x}_k) \to -\infty$ as $k \to -\infty$, then $q^* = -\infty$.

The solution $\boldsymbol{x}^*$ of the optimization problem (2.4) is called *optimal point*, if $\boldsymbol{x}^*$ is feasible and $f(\boldsymbol{x}^*) = q^*$. The set of all optimal values

$$\mathcal{X}_{\text{opt}} = \{\boldsymbol{x} \mid f(\boldsymbol{x}) = q^*,\ g_i(\boldsymbol{x}) \leq 0,\ i = 1, \ldots, m,\ h_j(\boldsymbol{x}) = 0,\ j = 1, \ldots, p\}$$

is called *optimal set*. The optimal value is *attained* (or *achieved*), if there exists an optimal point $\boldsymbol{x}^*$ for the problem (2.4), otherwise it is not attained (or achieved), i.e. the set $\mathcal{X}_{\text{opt}}$ is empty. If the optimal value is achieved then the optimization problem is *solvable*.

A feasible point $\boldsymbol{x}$ is *locally optimal* if it minimizes f in a subset of the feasible set, i.e. if there is an $R > 0$ such that

$$f(\boldsymbol{x}) = \inf\{f(\boldsymbol{z}) \mid g_i(\boldsymbol{z}) \leq 0,\ i = 1, \ldots, m,\ h_j(\boldsymbol{z}) = 0,\ j = 1, \ldots, p,\ ||\boldsymbol{z} - \boldsymbol{x}||_2 \leq R\}\,,$$

with $\boldsymbol{z} \in \mathbb{R}^n$, or equivalently, if it is the solution to the optimization problem

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{z}) & \\ \text{subject to} & g_i(\boldsymbol{z}) \leq 0, & i = 1, \ldots, m \\ & h_j(\boldsymbol{z}) = 0, & j = 1, \ldots, p \\ & ||\boldsymbol{z} - \boldsymbol{x}||_2 \leq R\,. & \end{array}$$

If a feasible point $\boldsymbol{x}$ minimizes f for the whole feasible set, then it is called *globally optimal*.

2.3.2 Convex Optimization Problems

An important class of optimization problems are *convex optimization problems*. These are of the form [17]

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{x}) & \\ \text{subject to} & g_i(\boldsymbol{x}) \leq 0, & i = 1, \ldots, m \\ & \boldsymbol{a}_j^T \boldsymbol{x} = \boldsymbol{b}_j, & j = 1, \ldots, p\,, \end{array} \tag{2.5}$$

where the objective function f and the inequality constraints functions $g_1, \ldots, g_m$ are convex, and the equality constraints functions are affine. Furthermore, the feasible set is convex; it is the intersection of the domain of the convex optimization problem (2.5), which is a convex set, with m convex sublevel sets $\{\boldsymbol{x} \mid g_i(\boldsymbol{x}) \leq 0,\ i = 1, \ldots, m\}$ and p hyperplanes $\{\boldsymbol{x} \mid \boldsymbol{a}_j^T \boldsymbol{x} = \boldsymbol{b}_j,\ j = 1, \ldots, p\}$, i.e.

$$\mathcal{O} = \mathbf{dom}\, f \cap \bigcap_{i=1}^{m} \mathbf{dom}\, g_i\,.$$

Based on the above a fundamental property of convex optimization problems is derived: any locally optimal point is also globally optimal.

2.3.3 Linear Optimization Problems

If the convex functions f and g_i of the optimization problem (2.5) are affine, then the problem is called *linear program* (LP). An LP can be formalized as [17]

$$\begin{aligned} &\text{minimize} && \boldsymbol{c}^T \boldsymbol{x} \\ &\text{subject to} && \boldsymbol{G}\boldsymbol{x} \preceq \boldsymbol{h} \\ & && \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\,, \end{aligned} \tag{2.6}$$

with $\boldsymbol{c} \in \mathbb{R}^n$, $\boldsymbol{G} \in \mathbb{R}^{m \times n}$, $\boldsymbol{h} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{p \times n}$, and $\boldsymbol{b} \in \mathbb{R}^p$.

Note that an LP can be solved efficiently, since it is P-hard, meaning that it can be solved in polynomial time[2].

2.3.4 Quadratic Optimization Problems

For the case where the objective function of the problem (2.6) is quadratic instead of affine, the optimization problem is called *quadratic program* (QP). A QP is described as [17]

$$\begin{aligned} &\text{minimize} && (1/2)\boldsymbol{x}^T \boldsymbol{Q}\boldsymbol{x} + \boldsymbol{p}^T \boldsymbol{x} \\ &\text{subject to} && \boldsymbol{G}\boldsymbol{x} \preceq \boldsymbol{h} \\ & && \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\,, \end{aligned} \tag{2.7}$$

where $\boldsymbol{Q} \in \mathbf{S}_+^n$, $\boldsymbol{p} \in \mathbb{R}^n$, $\boldsymbol{G} \in \mathbb{R}^{m \times n}$, $\boldsymbol{h} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{p \times n}$, and $\boldsymbol{b} \in \mathbb{R}^p$.

A QP can be either P-hard or NP-hard, depending on the matrix $\boldsymbol{Q}$. If $\boldsymbol{Q}$ is positive definite, then the QP is P-hard and can be solved in polynomial time. If $\boldsymbol{Q}$ is indefinite,

[2] An algorithm is said to be of polynomial running time if its time complexity is upper bounded by a polynomial expression in the length of the input n for the algorithm, i.e. $T(n) = O(n^c)$ for some constant $c > 0$. The reader may consult for additional and more detailed coverage of computational complexity theory on any of the available textbooks, such as [5, 50, 111].

or even if it has only one negative eigenvalue, then the QP is NP-hard, i.e. the solution time grows in exponential manner[3].

2.3.5 Mixed-Integer Linear Optimization Problems

The optimization variable in some cases may contain a continuous component and a binary part. The optimization problem (2.6) in this case is called *mixed-integer linear program* (MILP), and it is of the form [38]

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & \boldsymbol{G}\boldsymbol{x} \preceq \boldsymbol{h} \\ & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x}_b \in \{0,1\}^{n_b}\,, \end{array} \tag{2.8}$$

where $\boldsymbol{x} = [\boldsymbol{x}_r^T\ \boldsymbol{x}_b^T]^T$, with $\boldsymbol{x}_r \in \mathbb{R}^{n_r}$, $\boldsymbol{x}_b \in \{0,1\}^{n_b}$, and $n = n_r + n_b$. Furthermore, $\boldsymbol{c} \in \mathbb{R}^n$, $\boldsymbol{G} \in \mathbb{R}^{m\times n}$, $\boldsymbol{h} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{p\times n}$, and $\boldsymbol{b} \in \mathbb{R}^p$.

It should be noted that despite the fact that the objective function and the constraints functions are linear (or affine), the problem (2.8) is nonconvex because of the presence of the binary component. This means that the important property of convex optimization problems (see Section 2.3.2) does not apply to MILPs; the locally optimal points may not be globally optimal. Finally, an MILP is NP-hard, i.e. the running time depends exponentially on the number of the binary components.

2.3.6 Mixed-Integer Quadratic Optimization Problems

If the optimization variable of the problem (2.7) contain both a real-valued part and a binary part, i.e. it is of the form $\boldsymbol{x} = [\boldsymbol{x}_r^T\ \boldsymbol{x}_b^T]^T$, with $\boldsymbol{x}_r \in \mathbb{R}^{n_r}$, $\boldsymbol{x}_b \in \{0,1\}^{n_b}$, and $n = n_r + n_b$, then the formulated optimization problem is called *mixed-integer quadratic program* (MIQP) [37, 38]

$$\begin{array}{ll} \text{minimize} & (1/2)\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{p}^T\boldsymbol{x} \\ \text{subject to} & \boldsymbol{G}\boldsymbol{x} \preceq \boldsymbol{h} \\ & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x}_b \in \{0,1\}^{n_b}\,, \end{array} \tag{2.9}$$

with $\boldsymbol{Q} \in \mathbf{S}_+^n$, $\boldsymbol{p} \in \mathbb{R}^n$, $\boldsymbol{G} \in \mathbb{R}^{m\times n}$, $\boldsymbol{h} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{p\times n}$, and $\boldsymbol{b} \in \mathbb{R}^p$.

As already mentioned in Section 2.3.5, problem (2.9) is nonconvex because of the binary part $\boldsymbol{x}_b$, and it is NP-hard.

[3] An algorithm is said to be of exponential running time if its time complexity is upper bounded by $2^{\text{poly}(n)}$, where poly(n) is some polynomial in n, or in other words if $T(n) = O(2^{n^c})$ for some constant $c > 0$.

2.4 Enumeration Strategy for Mixed Integer Programming

In general, solving the mixed-integer optimization problems (MIPs) presented in Sections 2.3.5 and 2.3.6 is a very challenging task. For determining the solution of an MIP, either in the form of (2.8), or in the form of (2.9), for an MILP or an MIQP, respectively, a straightforward option is to use an enumeration strategy.

According to the *complete enumeration* method at each integer variable are progressively assigned the different values of its domain. The procedure is repeated until no more free integer variables are left, and the complete solution for the integer variables $\tilde{\boldsymbol{x}}_b$ is obtained [132]. Therefore, the MILP is simplified to an LP (or to a QP if the problem is an MIQP). By solving the resulting LP (or QP) the optimal value $q^* = f(\boldsymbol{x}^*)$ of the real-valued variables is determined.

Algorithm 2.1 Enumeration Strategy

function $f(x^*) =$ ENUM $(\boldsymbol{x}, \mathcal{S}, \mathcal{F})$
  $f(\boldsymbol{x}^*) = \infty$
  for $k = 1$ to n_b **do**
    select $i \in \mathcal{S}$
    $\mathcal{S} = \mathcal{S} \setminus \{i\}$; $\mathcal{F} = \mathcal{F} \cup \{i\}$
    for each $j \in \{0, 1\}$ **do**
      $\tilde{\boldsymbol{x}}_{b_i} = j$
      if $\mathcal{S} = \emptyset$ **then**
        $f(\boldsymbol{x}) = \inf\{\boldsymbol{c}^T\boldsymbol{x} \mid \boldsymbol{G}\boldsymbol{x} \preceq \boldsymbol{h}, \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x}_b = \tilde{\boldsymbol{x}}_b\}$
        if $f(\boldsymbol{x}) < f^*(\boldsymbol{x})$ **then**
          $f(\boldsymbol{x}^*) = f(\boldsymbol{x})$
        end if
      end if
    end for
  end for
  $\mathcal{S} = \{1, \ldots, n_b\}$; $\mathcal{F} = \emptyset$
end function

Based on the above the algorithm that describes the enumeration of the integer variables is shown in Algorithm 2.1[4]. For the convenience of the reader the form of the optimization variable $\boldsymbol{x}$ is recalled: $\boldsymbol{x} = [\boldsymbol{x}_r^T \ \boldsymbol{x}_b^T]^T$, with $\boldsymbol{x}_r \in \mathbb{R}^{n_r}$, $\boldsymbol{x}_b \in \{0, 1\}^{n_b}$, and $n = n_r + n_b$. Furthermore, the set $\mathcal{S}$ is the set of the free integer variables, i.e. in the first

[4]The presented algorithm is for an MILP. If the problem is an MIQP, then only line 9 needs to be modified, i.e. $f(\boldsymbol{x}) = \inf\{(1/2)\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{p}^T\boldsymbol{x} \mid \boldsymbol{G}\boldsymbol{x} \preceq \boldsymbol{h}, \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x}_b = \tilde{\boldsymbol{x}}_b\}$.

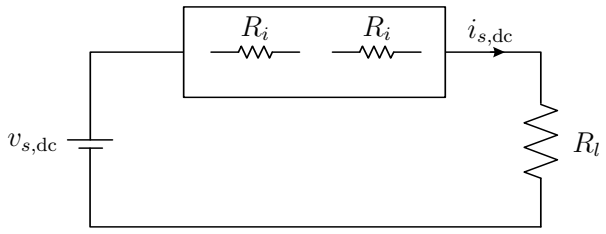


Figure 2.4: Linear circuit with resistive load of Example 2.2.

call of the algorithm $\mathcal{S} = \{1, \ldots, n_b\}$. Finally, the set $\mathcal{F}$ is the set of the fixed integer variables, and it is initially empty, $\mathcal{F} = \emptyset$.

As already mentioned, MIPs are computationally demanding (see Sections 2.3.5 and 2.3.6). The complete enumeration strategy requires a computational time that grows exponentially with the number of the integer variables $\boldsymbol{x}_b$, since the number of the examined operational modes grows in the same manner.

A way to speed-up the process of finding the optimal solution is to use *implicit enumeration*. The basic principle of implicit enumeration is to eliminate the infeasible solutions, and to evaluate, i.e. enumerate, only the feasible ones in order to find the best solution. Moreover, implicit enumeration can be done more efficiently with algorithms such as branch and bound, cutting plane, feasibility pump and others. However, details about these algorithms are beyond the scope of this thesis, but for further information the interested reader is referred to [18, 38].

Example 2.2. (Linear Circuit with Resistive Load): A resistive load of $R_l = 20\,\Omega$ is connected to a dc power supply $v_{s,\mathrm{dc}} = 10\,\mathrm{V}$. However, the current through it must be no more than $i_{s,\mathrm{dc}} = 400\,\mathrm{mA}$. Therefore, additional resistors must be connected in series (see Fig. 2.4). Resistors of four different values are available and they are the following

Table 2.1: Available Resistors (Example 2.2)

Resistance	Value (Ω)	Cost (€)
R_1	1.0	1.2
R_2	2.2	1.8
R_3	2.7	2.2
R_4	3.3	2.6

The goal is to keep the load current $i_{s,\mathrm{dc}}$ below its maximum allowable value, i.e. $i_{s,\mathrm{dc}} \leq 400\,\mathrm{mA}$, by adding some of the available resistors. Furthermore, the number of the additional resistors must be $N_{R,\mathrm{total}} \leq 3$, and the resulting total cost must be as low as possible.

Based on the above the problem can be formulated as an MILP. The decision of which resistors to select can be modeled with binary variables x_{b_i}, with $i = \{1, \ldots, n_b\}$

and $n_b = 4$; if resistor R_i is selected, then $x_i = 1$, while $x_i = 0$ if R_i is not selected. Furthermore, since there are no continuous-valued variables $n_r = 0$, thus, $n = n_b$. The resulting MILP is formulated as follows

$$\begin{aligned} \text{minimize} \quad & 1.2x_1 + 1.8x_2 + 2.2x_3 + 2.6x_4 \\ \text{subject to} \quad & -x_1 - 2.2x_2 - 2.7x_3 - 3.3x_4 \leq -5 \\ & x_1 + x_2 + x_3 + x_4 \leq 3 \\ & x_1, x_2, x_3, x_4 = 0 \text{ or } 1\,. \end{aligned} \tag{2.10}$$

The complete enumeration solution for this MILP is shown in Table 2.2.

Table 2.2: Possible Solutions of MILP (2.10)

Solution	x_1	x_2	x_3	x_4	Objective Function Value	Feasibility
1	0	0	0	0	∞	Infeasible
2	0	0	0	1	∞	Infeasible
3	0	0	1	0	∞	Infeasible
4	0	0	1	1	4.8	Feasible
5	0	1	0	0	∞	Infeasible
6	0	1	0	1	4.4	Feasible
7	0	1	1	0	∞	Infeasible
8	0	1	1	1	6.6	Feasible
9	1	0	0	0	∞	Infeasible
10	1	0	0	1	∞	Infeasible
11	1	0	1	0	∞	Infeasible
12	1	0	1	1	6	Feasible
13	1	1	0	0	∞	Infeasible
14	1	1	0	1	5.6	Feasible
15	1	1	1	0	5.2	Feasible
16	1	1	1	1	∞	Infeasible

As can be seen in Table 2.2, solutions $1-3, 5, 7, 9-11, 13, 16$ are infeasible. Solutions $1-3, 5, 7, 9-11, 13$ are eliminated because they violate the first constraint, while solution 16 is infeasible because it violates the second one. Therefore, by employing implicit enumeration only six solutions are taken into consideration. The best solution is achieved after evaluating the objective function value for these six solutions; the optimal one is solution 6 ($x_1 = 0,\ x_2 = 1,\ x_3 = 0,\ x_4 = 1$), which means that resistors R_2 and R_4 should be connected in series with the load. ■

2.5 Model Predictive Control

In the 1970s an advanced control methodology was developed in the process control industry—starting mainly from the petrochemical industry—known as *model predictive control* (MPC). Since then, the introduced control strategy has been gaining more widespread popularity. The reasons for this could be posited as being primarily twofold. The advent of faster microprocessors with increased computational capabilities[5], as well as the advantageous features of this optimal control strategy, enabled its application in many other sectors and industries, including the field of power electronics. Some of the most important features include its design simplicity due to the straightforward implementation procedure, the explicit inclusion of design criteria (constraints), and its ability to handle complex and nonlinear dynamics. Furthermore, in contrast to classical control methods, mainly based on the proportional-integral-derivative (PID) controller, MPC can be easily applied to a wider range of systems such as nonlinear plants, multi-input, multi-output (MIMO) plants, or input- and/or output-constrained plants[6].

In general, one could mention as the basic "components" of MPC the following:

1. *Mathematical model of the controlled plant*: The mathematical model of the system under investigation is required for the calculation of the evolution of the system states over time.

2. *Optimal control problem*: An objective function that embodies the control objectives is formulated. The optimization problem is solved, and the optimal sequence of control actions that results in the best behavior of the plant over the prediction horizon is derived. Note that prediction horizon is the time interval in which the control actions are planned, and the behavior of the plant is predicted.

3. *Receding horizon policy*: According to the receding horizon policy[7], only the first element of the optimal sequence of the control inputs is applied to the plant. The remaining elements are discarded, the prediction horizon is shifted forward by one sampling instant, and the optimization procedure is repeated.

[5] According to Moore's law [104] which states that the number of transistors on integrated circuits, and thus the computational power of microprocessors, doubles approximately every two years, the implementation of more demanding MPC algorithms, as well as their application in other scientific fields, will be possible in the forthcoming years.

[6] Classical PID-based control is suitable for linear, single-input, single-output (SISO), unconstrained problems. If the plant is nonlinear, MIMO, or constrained, then significant design effort is required. Nonlinearity can be overcome by gain scheduling, i.e. by, roughly speaking, designing controllers for different operating points and interpolate. MIMO systems have to be decomposed into SISO ones and then e.g. design cascaded control loops. Finally, anti-windup procedures are implemented to absorb different types of system constraints [6, 61].

[7] Model predictive control is also known as *receding horizon control* [81, 98].

In the next sections, the three components of MPC are explained in more detail. For further details on MPC, the reader is referred to [95, 99, 118].

2.5.1 Mathematical Model of the Controlled Plant

System Description: As the name of MPC implies, it is a model-based control strategy, therefore deriving an adequate model of the plant is prominent. Here, it should be mentioned that MPC is a discrete-time controller. Hence, hereafter the mathematical analysis of the model is restricted to the discrete-time domain.

The discrete-time dynamics of the system are of the form:

$$\boldsymbol{x}(k+1) = f\big(\boldsymbol{x}(k), \boldsymbol{u}(k)\big) \tag{2.11a}$$

$$\boldsymbol{y}(k) = g\big(\boldsymbol{x}(k)\big)\,, \tag{2.11b}$$

where $\boldsymbol{x}(k) \in \mathbb{R}^n$ is the state vector of the system at time instant kT_s, $\boldsymbol{u}(k) \in \mathbb{R}^m$ is the input vector at time instant kT_s, $\boldsymbol{y}(k) \in \mathbb{R}^p$ is the output vector at time instant kT_s, the functions f and g are the state-update and output functions, respectively, which can be linear or nonlinear, and T_s is the sampling interval.

The mathematical model (2.11) is used for the calculation of the state and output predictions. Starting from the current state $\boldsymbol{x}(k)$, and assuming a finite number N of planned control actions, i.e. $\{\boldsymbol{u}(k), \boldsymbol{u}(k+1), \ldots, \boldsymbol{u}(k+N-1)\}$, the state and the output of the plant at step $k+N$ can be evaluated. Therefore the state at step $k+1$ is

$$\boldsymbol{x}(k+1) = f\big(\boldsymbol{x}(k), \boldsymbol{u}(k)\big)\,. \tag{2.12}$$

At step $k+2$ the state is equal to

$$\boldsymbol{x}(k+2) = f\big(\boldsymbol{x}(k+1), \boldsymbol{u}(k+1)\big) = f\Big(f\big(\boldsymbol{x}(k), \boldsymbol{u}(k)\big), \boldsymbol{u}(k+1)\Big)\,. \tag{2.13}$$

Repeating the same procedure N times the state at step $k+N$ is given by

$$\begin{aligned} \boldsymbol{x}(k+N) &= f\big(\boldsymbol{x}(k+N-1), \boldsymbol{u}(k+N-1)\big) \\ &= f\Big(f\ldots\big(f\big(\boldsymbol{x}(k), \boldsymbol{u}(k)\big), \boldsymbol{u}(k+1)\big), \ldots, \boldsymbol{u}(k+N-1)\Big)\,. \end{aligned} \tag{2.14}$$

In the same way the output at step $k+N$ is

$$\begin{aligned} &\text{Step } k+1 \qquad \boldsymbol{y}(k+1) = g\big(\boldsymbol{x}(k+1)\big) = g\Big(f\big(\boldsymbol{x}(k), \boldsymbol{u}(k)\big)\Big) \\ &\text{Step } k+2 \qquad \boldsymbol{y}(k+2) = g\big(\boldsymbol{x}(k+2)\big) = g\Big(f\big(f(\boldsymbol{x}(k), \boldsymbol{u}(k)), \boldsymbol{u}(k+1)\big)\Big) \\ &\qquad\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ &\text{Step } k+N \qquad \boldsymbol{y}(k+N) = g\big(\boldsymbol{x}(k+N)\big) = g\Big(f\Big(f\ldots\big(f\big(\boldsymbol{x}(k), \boldsymbol{u}(k)\big), \boldsymbol{u}(k+1)\big), \\ &\qquad\qquad\qquad \ldots, \boldsymbol{u}(k+N-1)\Big)\Big)\,. \end{aligned} \tag{2.15}$$

Constraints: Apart from predicting the behavior of the state within the finite horizon, imposing constraints to the variables of concern is of fundamental importance. In this way the plant is able to operate at its physical limits without violating them. Thus, the most profitable operation can be achieved, while the operational limits of the plant are fully respected.

Constraints can be imposed to the state variables and/or to the manipulated variables, i.e. the control input. These variables can be upper, lower, or band bounded. The constraints that cannot be violated under any circumstances are called *hard constrains*; these that can be violated, but effort should be put into avoiding such violations are called *soft constrains*. Based on these constraints the feasible sets of the state and the control input are defined, i.e.

$$\mathcal{X} = \{\boldsymbol{x}(\ell) \mid \boldsymbol{c}_{\boldsymbol{x},l} \preceq \boldsymbol{x}(\ell) \preceq \boldsymbol{c}_{\boldsymbol{x},p},\ \ell = k, \ldots, k+N\}\,,$$

and

$$\mathcal{U} = \{\boldsymbol{u}(\ell) \mid \boldsymbol{c}_{\boldsymbol{u},l} \preceq \boldsymbol{u}(\ell) \preceq \boldsymbol{c}_{\boldsymbol{u},p},\ \ell = k, \ldots, k+N-1\}\,,$$

respectively. The vectors $\boldsymbol{c}_{\boldsymbol{x},l}, \boldsymbol{c}_{\boldsymbol{x},p} \in \mathbb{R}^n$ are the state lower and upper constraints, and $\boldsymbol{c}_{\boldsymbol{u},l}, \boldsymbol{c}_{\boldsymbol{u},p} \in \mathbb{R}^m$ are the input lower and upper constraints. If the ith state variable, with $i = 1, \ldots, n$, is upper bounded then $\boldsymbol{c}_{\boldsymbol{x},l_i} = -\infty$, and if it is lower bounded then $\boldsymbol{c}_{\boldsymbol{x},p_i} = +\infty$. Same case for the jth control signal, with $j = 1, \ldots, m$; if it is upper bounded then $\boldsymbol{c}_{\boldsymbol{u},l_j} = -\infty$, and if it is lower bounded then $\boldsymbol{c}_{\boldsymbol{u},p_j} = +\infty$.

Example 2.3. (Linear State-Space Model): Assume that there is a plant that can be described as a linear system. Furthermore, suppose that the state vector can be measured, and there is no measurement noise, or any other disturbances. The state-space model of the system is

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \tag{2.16a}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k)\,, \tag{2.16b}$$

According to (2.14), the evolution of the state of the model (2.16) within a N-step horizon is as follows

$$\begin{aligned}
\boldsymbol{x}(k+1|k) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \\
\boldsymbol{x}(k+2|k) &= \boldsymbol{A}\boldsymbol{x}(k+1|k) + \boldsymbol{B}\boldsymbol{u}(k+1|k) \\
&= \boldsymbol{A}^2\boldsymbol{x}(k) + \boldsymbol{A}\boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{B}\boldsymbol{u}(k+1|k) \\
&\vdots \qquad\qquad\qquad \vdots \\
\boldsymbol{x}(k+N|k) &= \boldsymbol{A}\boldsymbol{x}(k+N-1|k) + \boldsymbol{B}\boldsymbol{u}(k+N-1|k) \\
&= \boldsymbol{A}^N\boldsymbol{x}(k) + \boldsymbol{A}^{N-1}\boldsymbol{B}\boldsymbol{u}(k) + \ldots + \boldsymbol{B}\boldsymbol{u}(k+N-1|k)\,.
\end{aligned} \tag{2.17}$$

At step k the input vector depends on the *applied* input vector $\boldsymbol{u}(k-1)$, which is known. Therefore, the future control actions can be written with respect to the known input vector $\boldsymbol{u}(k-1)$ as

$$\begin{aligned} \boldsymbol{u}(k) &= \Delta\boldsymbol{u}(k) + \boldsymbol{u}(k-1) \\ \boldsymbol{u}(k+1|k) &= \Delta\boldsymbol{u}(k+1|k) + \Delta\boldsymbol{u}(k) + \boldsymbol{u}(k-1) \\ \vdots & \qquad\qquad \vdots \\ \boldsymbol{u}(k+N-1|k) &= \Delta\boldsymbol{u}(k+N-1|k) + \ldots + \Delta\boldsymbol{u}(k) + \boldsymbol{u}(k-1)\,, \end{aligned} \tag{2.18}$$

with $\Delta\boldsymbol{u}(k+i|k) = \boldsymbol{u}(k+i|k) - \boldsymbol{u}(k+i-1|k)$, and $i = 1, \ldots, N$.

Substituting (2.18) into (2.17) the result is

$$\begin{aligned} \boldsymbol{x}(k+1|k) =& \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\big(\Delta\boldsymbol{u}(k) + \boldsymbol{u}(k-1)\big) \\ \boldsymbol{x}(k+2|k) =& \boldsymbol{A}^2\boldsymbol{x}(k) + \boldsymbol{A}\boldsymbol{B}\big(\Delta\boldsymbol{u}(k) + \boldsymbol{u}(k-1)\big) + \\ &+ \boldsymbol{B}\big(\Delta\boldsymbol{u}(k+1|k) + \Delta\boldsymbol{u}(k) + \boldsymbol{u}(k-1)\big) \\ =& \boldsymbol{A}^2\boldsymbol{x}(k) + (\boldsymbol{A}+\boldsymbol{I})\boldsymbol{B}\Delta\boldsymbol{u}(k) + \boldsymbol{B}\Delta\boldsymbol{u}(k+1|k) + (\boldsymbol{A}+\boldsymbol{I})\boldsymbol{B}\boldsymbol{u}(k-1) \\ \vdots & \qquad\qquad\qquad\qquad \vdots \\ \boldsymbol{x}(k+N|k) =& \boldsymbol{A}^N\boldsymbol{x}(k) + (\boldsymbol{A}^{N-1} + \ldots + \boldsymbol{A} + \boldsymbol{I})\boldsymbol{B}\Delta\boldsymbol{u}(k) + \\ &+ \ldots + \boldsymbol{B}\Delta\boldsymbol{u}(k+N-1|k) + (\boldsymbol{A}^{N-1} + \ldots + \boldsymbol{A} + \boldsymbol{I})\boldsymbol{B}\boldsymbol{u}(k-1)\,. \end{aligned} \tag{2.19}$$

Equivalently (2.19) can be written in a matrix form

$$\begin{aligned} \begin{bmatrix} \boldsymbol{x}(k+1|k) \\ \vdots \\ \boldsymbol{x}(k+N|k) \end{bmatrix} = & \underbrace{\begin{bmatrix} \boldsymbol{A} \\ \vdots \\ \boldsymbol{A}^N \end{bmatrix} \boldsymbol{x}(k) + \begin{bmatrix} \boldsymbol{B} \\ \vdots \\ \sum_{i=0}^{N-1} \boldsymbol{A}^i\boldsymbol{B} \end{bmatrix} \boldsymbol{u}(k-1) +}_{\text{past}} \\ & + \underbrace{\begin{bmatrix} \boldsymbol{B} & \cdots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{N-1} \boldsymbol{A}^i\boldsymbol{B} & \cdots & \boldsymbol{B} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{u}(k) \\ \vdots \\ \Delta\boldsymbol{u}(k+N-1|k) \end{bmatrix}}_{\text{future}} \end{aligned} \tag{2.20}$$

The predictions of the output are obtained in the same manner

$$\begin{aligned} \boldsymbol{y}(k+1|k) &= \boldsymbol{C}\boldsymbol{x}(k+1|k) \\ \boldsymbol{y}(k+2|k) &= \boldsymbol{C}\boldsymbol{x}(k+2|k) \\ \vdots & \qquad \vdots \\ \boldsymbol{y}(k+N|k) &= \boldsymbol{C}\boldsymbol{x}(k+N|k)\,, \end{aligned} \tag{2.21}$$

or in a matrix form

$$\begin{bmatrix} \boldsymbol{y}(k+1|k) \\ \vdots \\ \boldsymbol{y}(k+N|k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{C} & \cdots & \boldsymbol{0} \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \boldsymbol{C} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(k+1|k) \\ \vdots \\ \boldsymbol{x}(k+N|k) \end{bmatrix} \tag{2.22}$$

■

2.5.2 Optimal Control Problem

Control Objectives: For the formulation of the optimal control problem the control objectives should be clearly defined. Based on these an objective function is designed as follows

$$J\big(\boldsymbol{x}(k), \boldsymbol{U}(k)\big) = \sum_{\ell=k}^{k+N-1} P\big(\boldsymbol{x}(\ell+1|k), \boldsymbol{u}(\ell|k)\big)\,. \tag{2.23}$$

In (2.23) P is a stage cost function based on the p-norm[8]. Function P may take into account the deviation of the output variables from their reference values, or the required control effort. Hence, by applying a sequence of control moves $\boldsymbol{U}(k) = [\boldsymbol{u}(k)^T\ \boldsymbol{u}(k+1)^T\ \ldots\ \boldsymbol{u}(k+N-1)^T]^T$, the objective function (2.23) penalizes the evolution of the output error and the overall control effort over the horizon N.

Finite Time Optimal Control Problem: The goal of the optimization problem is to find the sequence of control moves $\boldsymbol{U}(k)$ that results in the best, i.e. *optimal*, performance of the plant. This is translated as

$$\begin{aligned} &\text{minimize} && J(k) \\ &\text{subject to} && \boldsymbol{x}(\ell+1) = f\big(\boldsymbol{x}(\ell), \boldsymbol{u}(\ell)\big) \\ & && \boldsymbol{x}(\ell) \in \mathcal{X},\ \ell = k, \ldots, k+N \\ & && \boldsymbol{u}(\ell) \in \mathcal{U},\ \ell = k, \ldots, k+N-1\,. \end{aligned} \tag{2.24}$$

This yields the optimal control input sequence at step k, $\boldsymbol{U}^*(k) = [\boldsymbol{u}^*(k)^T\ \boldsymbol{u}^*(k+1)^T\ \ldots\ \boldsymbol{u}^*(k+N-1)^T]^T$. It is important to note that, as the nature of the problem (2.24) implies, the optimal control input sequence $\boldsymbol{U}^*(k)$

$$\boldsymbol{U}^*(k) = \arg\min_{\boldsymbol{U}(k)} J\big(\boldsymbol{x}(k), \boldsymbol{U}(k)\big)\,, \tag{2.25}$$

is an *open-loop* solution.

[8] Usually in MPC the 1-, 2-, or ∞-norm are used. If $p = 1$, or $p = \infty$, then a linear objective function results; if $p = 2$ a quadratic one.

2.5.3 Receding Horizon Policy

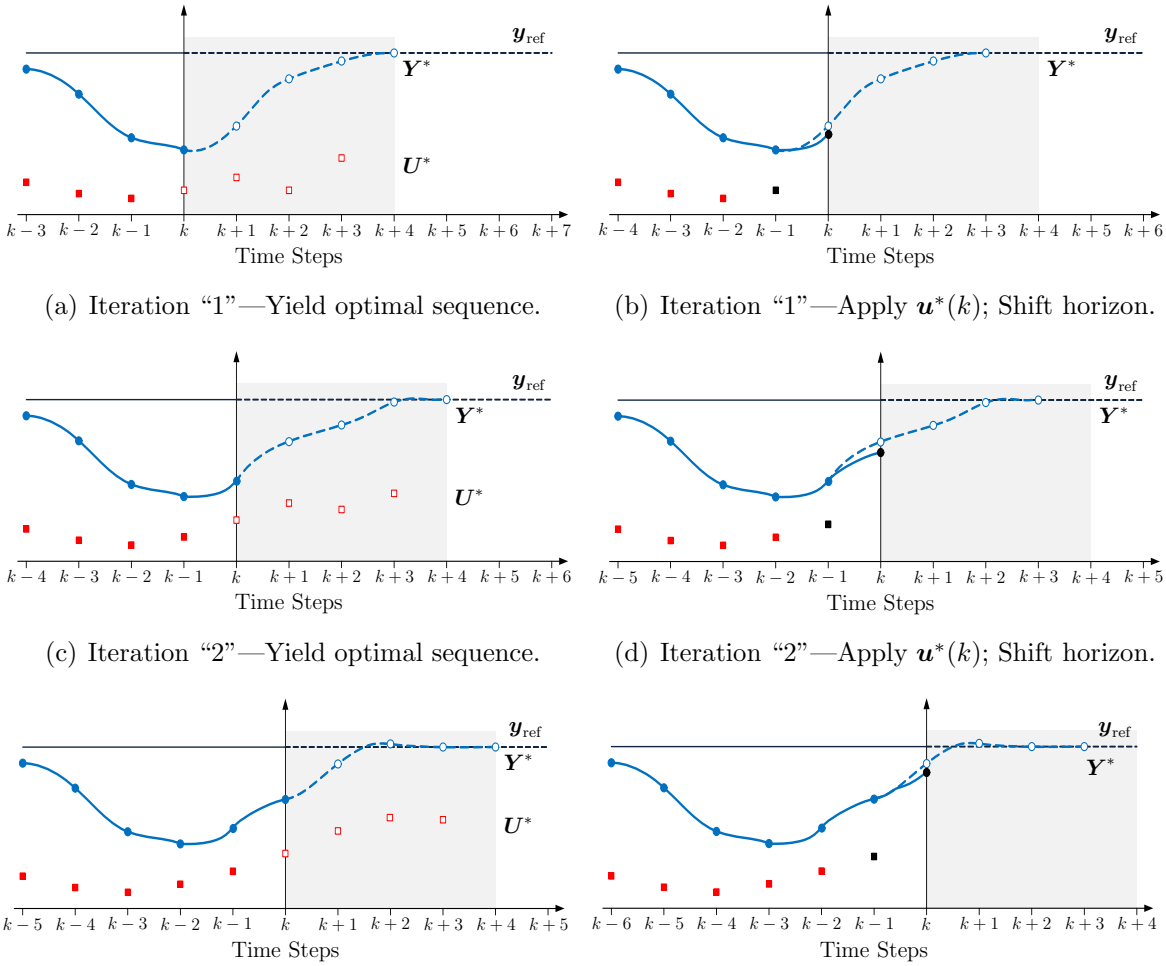
As mentioned in Section 2.5.2, the derived optimal control input sequence $\boldsymbol{U}^*(k)$ is a solution to the open-loop optimization problem (2.24). To provide feedback, allowing one to cope with plant uncertainties and disturbances, the so-called *receding horizon policy* is employed, whereby only the first input of the optimal sequence $\boldsymbol{u}^*(k)$ is used as the process input. At the next step, k is set to $k+1$, new state measurements or estimates are obtained, the horizon is shifted by one sampling interval and the optimization problem (2.24) is solved again.

Combining the three "ingredients" of MPC, i.e. the mathematical model of the controlled plant, the optimal control problem, and the receding horizon policy, an MPC algorithm comprises the following steps:

1. Obtain state measurements and/or estimates.
2. Solve optimization problem (2.24) based on $\boldsymbol{x}(k)$.
3. Obtain the optimal control input sequence $\boldsymbol{U}^*(k) = [\boldsymbol{u}^*(k)^T\ \boldsymbol{u}^*(k+1)^T\ \ldots\ \boldsymbol{u}^*(k+N-1)^T]^T$.
4. Apply $\boldsymbol{u}^*(k)$ to the plant.
5. Set $k = k+1$.
6. Go back to step "1".

Example 2.4. (Model Predictive Control): Let a plant which can be described by (2.11). The control objective is to achieve output-reference tracking, i.e. to get $\boldsymbol{y} = \boldsymbol{y}_{\text{ref}}$. To do so MPC is employed; an objective function is formulated based on the aforementioned control objective, and the evolution of the output is calculated within a four-step prediction horizon ($N = 4$).

In order to visualize the MPC procedure for this case, an illustrative example is presented in Fig. 2.5. At time step k the optimization problem is solved, yielding an optimal sequence of control moves $\boldsymbol{U}^*(k)$. The segment of interest is depicted in Fig. 2.5(a). The optimal control inputs $\boldsymbol{U}^*(k)$ within the four-step horizon, indicated with red empty squares, and the resulting trajectory of the output variable, indicated with blue empty circles connected with a blue dashed line, are shown. Furthermore, the applied control inputs are shown as red solid squares, and the corresponding values of the output variable are shown as blue solid circles connected with a blue solid line. Finally, the past output reference trajectory is indicated as a solid magenta line, while the future reference trajectory as a dashed one. Out of this sequence only the first element is applied, now shown as a black solid square, and corresponds to step $k-1$ (Fig. 2.5(b)). Note that



(a) Iteration "1"—Yield optimal sequence.

(b) Iteration "1"—Apply $\boldsymbol{u}^*(k)$; Shift horizon.

(c) Iteration "2"—Yield optimal sequence.

(d) Iteration "2"—Apply $\boldsymbol{u}^*(k)$; Shift horizon.

(e) Iteration "3"—Yield optimal sequence.

(f) Iteration "3"—Apply $\boldsymbol{u}^*(k)$; Shift horizon.

Figure 2.5: An example of a model predictive control algorithm with a four-step prediction horizon. Six snapshots from three iterations of the controller are shown. The future and applied optimal control inputs are indicated as red squares, empty and solid, respectively. The future and past values of the output are indicated as blue circles, empty and solid, respectively. The blue line that passes through them shows the output trajectory; the solid line corresponds to the past trajectory and the dashed to the predicted one. The magenta line shows the output reference trajectory; the solid line corresponds to the past output reference trajectory and the dashed to the future one. Finally, the control input that has just been applied $\boldsymbol{u}^*(k-1)$ is shown as a black solid square, and the resulting output $\boldsymbol{y}^*(k)$ as a black solid circle.

the resulting output variable $\boldsymbol{y}^*(k)$, indicated as a black solid circle, is not the same as the predicted one (shown as blue empty circle). This mismatch may arise due to several reasons, such as mismatches between the mathematical model of the plant, used as prediction model, and the actual model of the plant (e.g. due to assumptions that have been made in the modeling process), model uncertainties, measurement noise, quantization error, etc. Finally, as it can be observed, the discrete time updates ($k = k + 1$), and the

prediction horizon is shifted by one sampling interval.

In a next step, the optimization problem is solved again over the shifted horizon; in Fig. 2.5(c) the new optimal sequence of control inputs $\boldsymbol{U}^*(k)$ (shown as red empty squares), as well as the resulting output trajectory are shown (blue dashed line that passes through the discrete values of $\boldsymbol{Y}^*(k)$ indicated as blue empty circles). Once again, only the first element is applied (black solid square—see Fig. 2.5(d)), while the output variable $\boldsymbol{y}^*(k)$ (black solid circle) is different from the predicted one (blue empty circle), because of possible mismatches mentioned above. Furthermore, the horizon is shifted by one sampling interval before the next iteration of the controller.

Eventually, by repeating the optimization procedure over and over again the output will track its reference (in Figs. 2.5(e) and 2.5(f) the results for one more iteration are presented). ■

2.5.4 Enumeration-Based Model Predictive Control

Algorithm 2.2 MPC General Algorithm

function $\boldsymbol{u}^*(k)$ = MPC $(\boldsymbol{x}(k), \boldsymbol{u}(k-1))$
  $J^*(k) = \infty$; $\boldsymbol{u}^*(k) = \emptyset$
  for all $\boldsymbol{U}$ over N **do**
    $J = 0$
    $i = 1$
    for $\ell = k$ to $k + N - 1$ **do**
      $\boldsymbol{x}(\ell + 1|k) = f\big(\boldsymbol{x}(\ell|k), \boldsymbol{u}(\ell|k)\big)$
      $J(i) = P\big(\boldsymbol{x}(\ell + 1|k), \boldsymbol{u}(\ell|k)\big)$
      $J = J + J(i)$
      $i = i + 1$
    end for
    if $J < J^*(k)$ **then**
      $J^*(k) = J$, $\boldsymbol{u}^*(k) = \boldsymbol{U}(1)$
    end if
  end for
end function

When MPC schemes are used to control discrete-time hybrid systems, the formulated optimization problem is an MIP, either in the form (2.8), or in the form (2.9). Thereby, using implicit enumeration—as presented in Section 2.4—to solve (2.24) is an alternative.

By considering all possible combinations of the control input and the current state, the state at the next time-instant and the output are computed based on the hybrid

model of the plant. This procedure is repeated N times within a N-step horizon, as explained in Section 2.5.1. Thus, given all possible sequences of the control moves $\boldsymbol{U}(k) = [\boldsymbol{u}(k)^T\ \boldsymbol{u}(k+1)^T\ \ldots\ \boldsymbol{u}(k+N-1)^T]^T \in \mathcal{U}^N$, and the initial state $\boldsymbol{x}(k) \in \mathcal{X}$ the state evolution is predicted over the horizon.

Based on the above, the enumeration-based MPC algorithm is as the one presented in Section 2.5.3, with the difference that steps 2 and 3 are replaced by the new step 2, shown below.

1. Obtain state measurements and/or estimates.
2. Execute Algorithm 2.2.
3. Apply $\boldsymbol{u}^*(k)$ to the plant.
4. Set $k = k + 1$.
5. Go back to step "1".

Part II

Applications

Chapter 3

DC-DC Converters

3.1 Introduction

Dc-dc converters are electronic circuits which transfer energy from a dc source to a load. By appropriately manipulating the switches of a dc-dc converter the input energy is temporarily stored and then released; in this way the converter—depending on its type—can produce an output with smaller or larger magnitude, or even with reversed polarity compared to the input dc voltage.

Over the past decades dc-dc conversion has matured into a ubiquitous technology, which is used in a wide variety of applications, including power supplies for computers, portable electronic devices, battery chargers, and dc motor drives. This is due to the fact that dc-dc converters are circuits with small size, light weight, high power density, and high efficiency.

In their simplest form dc-dc converters comprise two semiconductor switches that are periodically switched on and off, and a low-pass filter with an inductor and a capacitor. The filter is added to pass the dc component of the input, and to remove the switching harmonics, and, thus, to produce at the output a dc voltage with a small ripple. Usually, out of the two switches only one is controllable, while the other is dually operated. However, more complex topologies have been introduced in the last years that use two bidirectional controllable switches.

Despite the fact that the switch-mode dc-dc conversion is a well-established technology, the problems associated with these applications and their closed-loop controlled performance still pose theoretical and practical challenges. An appropriate control strategy should achieve the regulation of the output voltage of the converter to a desired value despite changes in the input voltage and the load, since such variations are very common; in many cases the input voltage is unregulated, e.g. when a rectifier and a dc-dc converter are connected in cascade, or the load is time-varying or nonlinear.

Dc-dc converters are intrinsically difficult to control due to their switching behavior, constituting a (continuous-time) switched linear or *hybrid* system. In particular, depending on the position of the switches and the value of the current, there are three different operating modes, each one governed by different linear continuous-time dynamical laws. Furthermore, constraints that stem from the topology of the converter are present. For example, the duty cycle[1] is bounded between zero and one, while the current through the inductor cannot be negative. Finally other constraints, such as an upper limit on the current during start-up for a soft start, can be imposed.

Based on the above, it is evident that a controller should turn on and off the controllable switch such that the output voltage becomes equal to its reference value. In general, this is achieved with pulse width modulation (PWM) techniques. Hence, by conforming the pulse width, i.e. by modifying the duty cycle d, the output voltage is regulated to the desired level. However, there are strategies where a modulator is not required. According to these methods the switch is directly manipulated, i.e. control signals are sent directly to the switch, without the presence of an intermediate modulator[2]. Regardless of the methodology employed—with or without a modulator—the control problem is to decide when the switch is to be turned on and off.

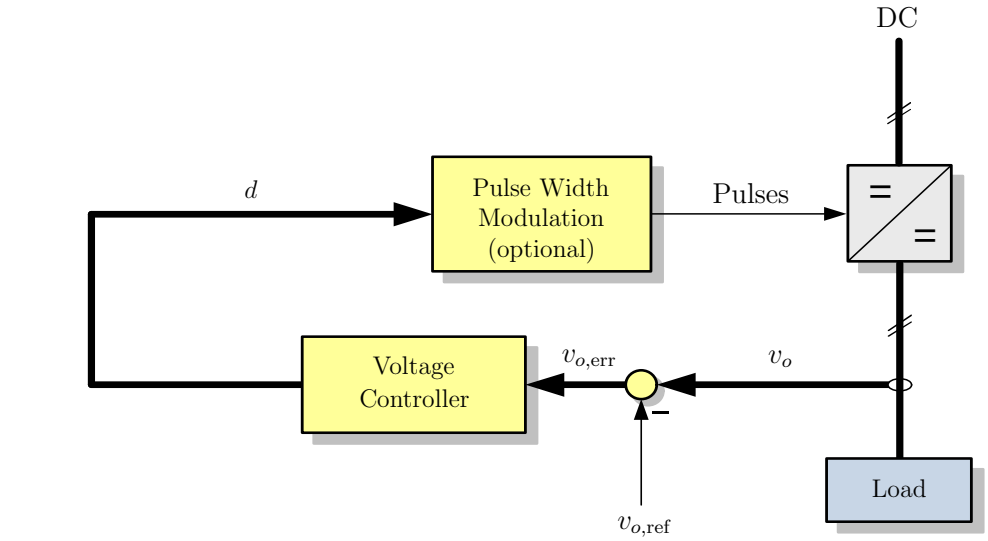
3.1.1 Control of DC-DC Converters

As already mentioned, many of the difficulties in controlling dc-dc converters arise from their hybrid nature. To bypass these obstacles, the modeling of the converter is based on state-space averaging [35]. With this modeling approach only the important dominant behavior of the plant is modeled, while other small but complicating phenomena are neglected. Therefore, a mathematical model of the converter is derived that uses the duty cycle as the system input. However, due to approximations made during the design process, such as that the modulation frequency is much smaller than converter switching frequency, only the slow dynamics of the system are modeled; only the basic insight is gained, since the switching nature of the system is ignored. Thereby, with the averaging approach all information about the fast dynamics of the system is lost.

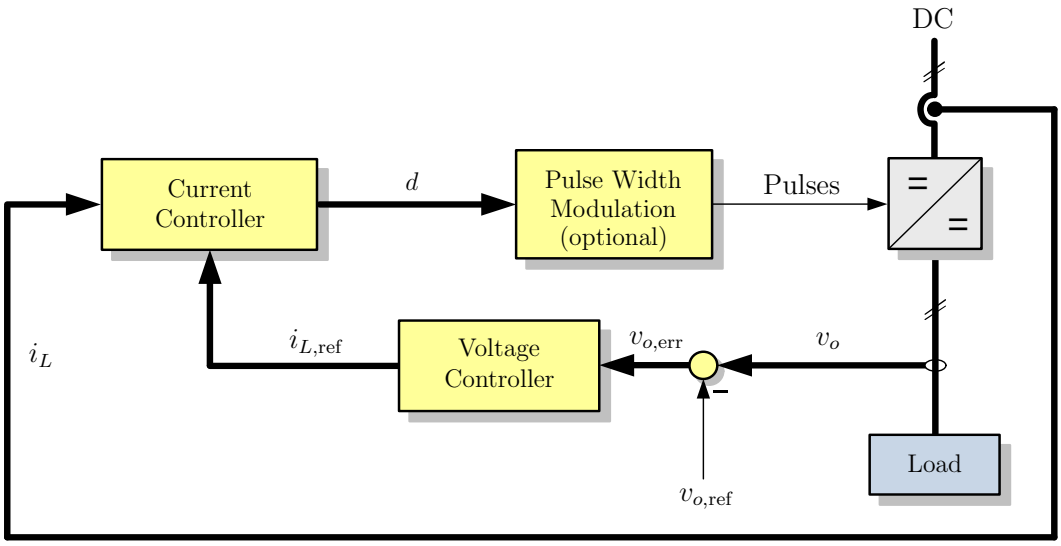
The derived continuous-time mathematical model is nonlinear since the state variables are multiplied with the duty cycle. In order to simplify the controller design procedure, the nonlinear average model is linearized around a specific operating point. Nevertheless, the (linear) controllers carried out with this procedure are usually tuned to achieve optimal performance only over a narrow operating range; outside this range the performance is significantly deteriorated.

[1]Duty cycle d is the percent of time that the switch remains in an active state t_{on} as a fraction of the switching period T_{sw} ($T_{sw} = 1/f_{sw}$, where f_{sw} is the switching frequency), i.e. $d = \frac{t_{on}}{T_{sw}}$.

[2]Note that the absence of the modulator results in a variable switching frequency.



(a) Voltage control.



(b) Current control.

Figure 3.1: General control block diagrams of dc-dc converters.

An alternative to classic state-space averaging techniques is to model the system in the discrete-time domain. In this way, a nonlinear model (with respect to the control input) results that captures all the complex dynamics of the converter, while inter- and subharmonics that occur during the closed-loop operation of the system are fully analyzed. The major disadvantage of this approach is that the controller design procedure requires a significant effort since a nonlinear model of the plant is used.

For the closed-loop operation of dc-dc converters several control techniques have been proposed, which can be divided into two main groups: voltage-mode and current-mode controllers (Fig. 3.1) [36]. In the first category, the control objective is the elimination

of the voltage error, i.e. the difference between the measured output voltage and the reference value. This is typically achieved by employing a single loop that directly controls the voltage, see Fig. 3.1(a). The voltage control problem is difficult, since it relates to a second order system with a nonminimum phase behavior, i.e. the control-to-output voltage transfer function contains a right half-plane zero, resulting in a reverse-response system behavior during transients [3, 28].

In contrast to that, current-mode controllers employ two loops (Fig. 3.1(b)). The outer loop constitutes the voltage regulation loop, which manipulates the current reference so as to remove any output voltage error. The inner loop is the current regulation loop, which controls the measured or estimated inductor current along its reference. The switching state is typically manipulated indirectly via a modulator using the notion of the duty cycle. Despite the fact that for current-mode controllers two loops are required, this type of controllers is more often employed since the design procedure is simpler; the current exhibits a minimum phase behavior with respect to the control action (and it is a first-order system).

In literature many different approaches to the control problem can be found. These schemes could be divided into two main groups: the linear and the nonlinear controllers. Furthermore, an additional classification would be based on the mathematical model of the converter used. Hence, there exist controllers designed based on the average and non-average plant model.

The majority of the controllers are based on the conventional PI controller. These schemes are tuned on the basis of the linear state-space average model of the converter. The design procedure is trivial: a crossover frequency is selected to be an order of magnitude smaller than the switching frequency, while the phase margin should be between 45° and 60° [3, 35].

In [88] and [89] a linear quadratic regulator (LQR) is proposed. The controller is based on the locally linearized discrete-time averaged model. In addition, an outer estimation loop that effectively adds an integrator is employed. Nonetheless, the limitations stem from the linear nature of the controller are present; the operating range is limited since the model used is only locally linearized, while constraints cannot be handled.

Throughout the years, several nonlinear controllers based on the averaged or nonaveraged state-space model of the converter have been proposed, as well. Controllers based on fuzzy logic [51, 97] and feedforward control [71, 72] make use of the averaged model. However, in these works the converter is considered lossless. In [122] the author design a family of PI controllers that depend nonlinearly on the control input, i.e. the duty cycle. The tuning of the controllers is done according to the guidelines of Ziegler and Nichols. However, constraints on the control input are not tackled, e.g. by implementing anti-windup procedures as explained in Section 2.5, while the performance of the overall

control scheme is somewhat deteriorated because of the presence of a low-pass filter in the outer loop. What is noteworthy in [80] is that a nonlinear H_∞ controller is proposed, the closed-loop stability of which is verified via Lyapunov function. In [96] a sliding mode controller as a current-mode controller is designed. A sliding surface is used for the average inductor current; by keeping the inductor current close to this surface the output voltage is indirectly controlled. Furthermore, stability and effects of controller parameter variations are investigated. Finally, in [128] a detailed overview of sliding mode controllers for dc-dc converters is given, while implementation related issues are addressed.

Although the aforementioned approaches have been shown to be reasonably effective, several challenges have not been fully addressed yet, such as ease of controller design and tuning, as well as robustness to load parameter variations. Moreover, the aim, not only to improve the performance of the closed-loop system, but to also enable a systematic design and implementation procedure, still exists. Furthermore, the recent theoretical advances with regards to controlling hybrid systems, as well as the emergence of fast microprocessors that enabled the implementation of more computationally demanding algorithms, allow one to tackle these problems in a novel way. Since MPC is a particularly promising candidate to fulfill all these goals, several algorithms have been proposed the recent years.

MPC has been typically used in its simplest form—namely as a dead-beat controller—for controlling the predominant dc-dc converter topologies, i.e. the buck, the boost and the buck-boost converter [16,26,27,117,137]. A more complex MPC strategy was introduced in [45, 47] for the buck, and in [7, 8] for the boost converter. The nonlinear dynamics of the converter were approximated by a piecewise affine (PWA) model; the resulting controller regions were computed offline and stored in a look-up table, greatly reducing the computation time required to solve the control problem in real-time. In [107] an MPC approach based on numerical techniques was presented, and a sliding mode observer was designed, providing estimates of the varying voltage source and load resistor. In [136] the control problem of a full bridge dc-dc converter was formulated in the context of MPC in a computationally efficient manner.

In this work, MPC is employed as both a current- and a voltage-mode controller. A discrete-time model of the converter is introduced, which captures all operating modes of the inductor current, making it suitable for operation both in the continuous (CCM) and discontinuous conduction mode (DCM). For both MPC-based schemes enumeration is used, i.e. all the possible switching transitions are evaluated in real-time. However, depending on the nature of the controller (current or voltage controller), different strategies to tackle the inherent increased computational complexity are presented. Furthermore, a state estimation scheme is implemented that addresses load uncertainties and model

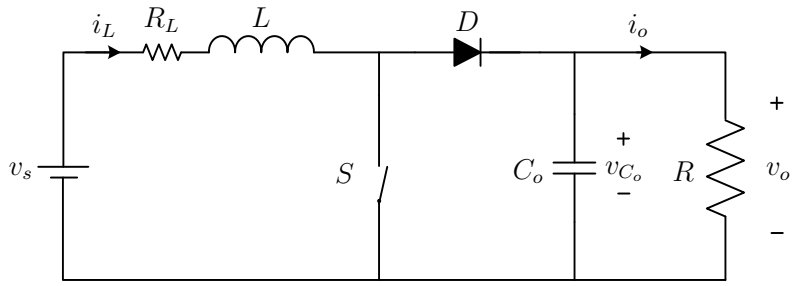


Figure 3.2: Topology of the dc-dc boost converter.

mismatches.

3.2 Model of the Boost Converter

3.2.1 Continuous-Time Model

As already mentioned, the dc-dc boost converter, shown in Fig. 3.2, is a converter that increases the (typically uncontrolled) dc input voltage $v_s(t)$ to a higher (controlled) dc output voltage $v_o(t)$. The converter consists of two power semiconductors—the controllable switch S, and the diode D. The inductor L with the internal resistor R_L is used to store and deliver energy depending on the operating mode of the converter, while the filter capacitor C_o is connected in parallel with the load resistor R so as to ensure a constant output voltage during steady-state operation of the converter.

The converter can operate in continuous (CCM) and discontinuous (DCM) conduction mode, depending on the value of the inductor current $i_L(t)$, see Fig. 3.3. Three different linear dynamics are associated with the switch positions that capture all operating modes of the inductor current. When the switch S is *on* ($S = 1$), energy is stored in the inductor L and the inductor current $i_L(t)$ increases. When the switch S is *off* ($S = 0$), the inductor is connected to the output and energy is released through it to the load, resulting in a decreasing $i_L(t)$. Furthermore, when the switch S remains *off* and $i_L(t) = 0$, then both S and D are *off*; the topology is reduced to the mesh formed by the capacitor C_o and the load. In this case, the converter operates in DCM.

The state-space representation of the converter in the continuous-time domain is given by the following equations [36]

$$\frac{d\boldsymbol{x}(t)}{dt} = \big(\boldsymbol{A}_1 + \boldsymbol{A}_2\boldsymbol{u}(t)\big)\boldsymbol{x}(t) + \boldsymbol{B}v_s(t) \tag{3.1a}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)\,, \tag{3.1b}$$

where

$$\boldsymbol{x}(t) = \begin{bmatrix} i_L(t) & v_o(t) \end{bmatrix}^T, \tag{3.2}$$

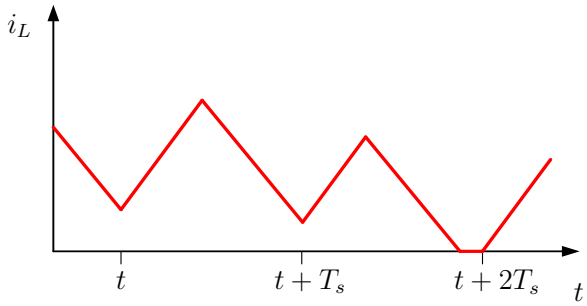


Figure 3.3: The shape of the inductor current reveals the operation mode: the converter operates in CCM from t to $t+T_s$, and in DCM from $t+T_s$ to $t+2T_s$.

is the state vector, encompassing the inductor current and the output voltage across the output capacitor. The output

$$\boldsymbol{y}(t) = v_o(t) \tag{3.3}$$

is given by the output voltage. The system matrices are

$$\boldsymbol{A}_1 = \begin{bmatrix} -\frac{d_{\text{aux}}R_L}{L} & -\frac{d_{\text{aux}}}{L} \\ \frac{d_{\text{aux}}}{C_o} & -\frac{1}{C_oR} \end{bmatrix}, \ \boldsymbol{A}_2 = \begin{bmatrix} 0 & \frac{1}{L} \\ -\frac{1}{C_o} & 0 \end{bmatrix}, \ \boldsymbol{B} = \begin{bmatrix} \frac{d_{\text{aux}}}{L} & 0 \end{bmatrix}^T, \text{ and } \boldsymbol{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

The variable $\boldsymbol{u}$ denotes the switch position, with $u = 1$ implying that the switch S is *on*, and $u = 0$ referring to the case where the switch S is *off*. Finally, d_{aux} is an auxiliary binary variable [12] that is $d_{\text{aux}} = 1$ when the converter operates in CCM, i.e. either $u = 1$ or $u = 0$ and $i_L(t) > 0$. When the converter operates in DCM, i.e. $u = 0$ and $i_L(t) = 0$, then $d_{\text{aux}} = 0$ holds.

$$d_{\text{aux}}(t) = \begin{cases} 1 \text{ if } u(t) = 1, \text{ or } u(t) = 0 \text{ and } i_L(t) > 0 \\ 0 \text{ if } u(t) = 0 \text{ and } i_L(t) = 0 \end{cases} \tag{3.4}$$

For a graphical summary, representing the boost converter as an automaton, see Fig. 3.4.

Therefore, by setting $d_{\text{aux}} = 0$ or $d_{\text{aux}} = 1$, depending on the operating mode, state-space model of the converter can be rewritten as

$$\frac{d\boldsymbol{x}(t)}{dt} = \begin{cases} \boldsymbol{\Gamma}_1\boldsymbol{x}(t) + \boldsymbol{\Delta}v_s(t) & S = 1 \\ \boldsymbol{\Gamma}_2\boldsymbol{x}(t) + \boldsymbol{\Delta}v_s(t) & S = 0 \ \& \ i_L(t) > 0 \\ \boldsymbol{\Gamma}_3\boldsymbol{x}(t) & S = 0 \ \& \ i_L(t) = 0 \end{cases} \tag{3.5a}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)\,, \tag{3.5b}$$

where now the matrices $\boldsymbol{\Gamma}_1$, $\boldsymbol{\Gamma}_2$, $\boldsymbol{\Gamma}_3$ and $\boldsymbol{\Delta}$ are given by

$$\boldsymbol{\Gamma}_1 = \begin{bmatrix} -\frac{R_L}{L} & 0 \\ 0 & -\frac{1}{C_oR} \end{bmatrix}, \ \boldsymbol{\Gamma}_2 = \begin{bmatrix} -\frac{R_L}{L} & -\frac{1}{L} \\ \frac{1}{C_o} & -\frac{1}{C_oR} \end{bmatrix}, \ \boldsymbol{\Gamma}_3 = \begin{bmatrix} 0 & 0 \\ 0 & -\frac{1}{C_oR} \end{bmatrix}, \text{ and } \boldsymbol{\Delta} = \begin{bmatrix} \frac{1}{L} & 0 \end{bmatrix}^T.$$

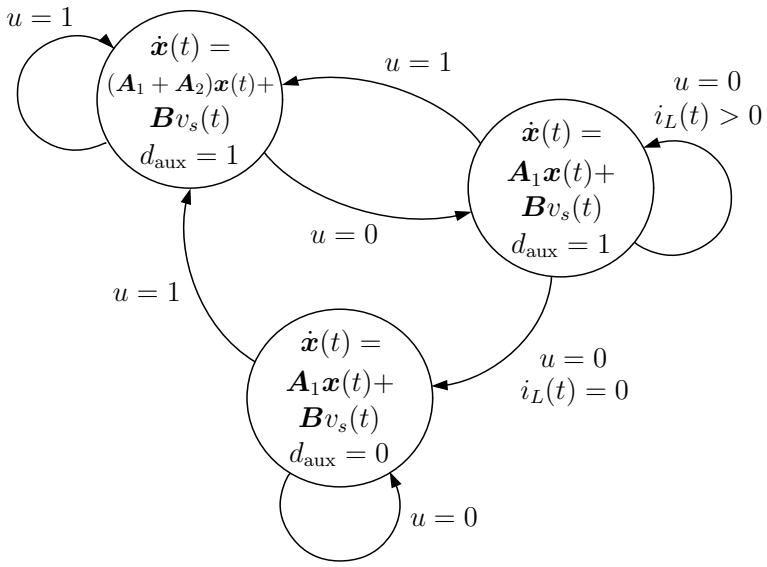


Figure 3.4: Dc-dc converter presented as a continuous-time automaton.

3.2.2 Discrete-Time Model

The derivation of an adequate model of the boost converter to serve as an internal prediction model for MPC is of fundamental importance. As can be seen in Fig. 3.5, after the discretization of the model in time, the converter can operate in four different modes, depending on the shape of the inductor current:

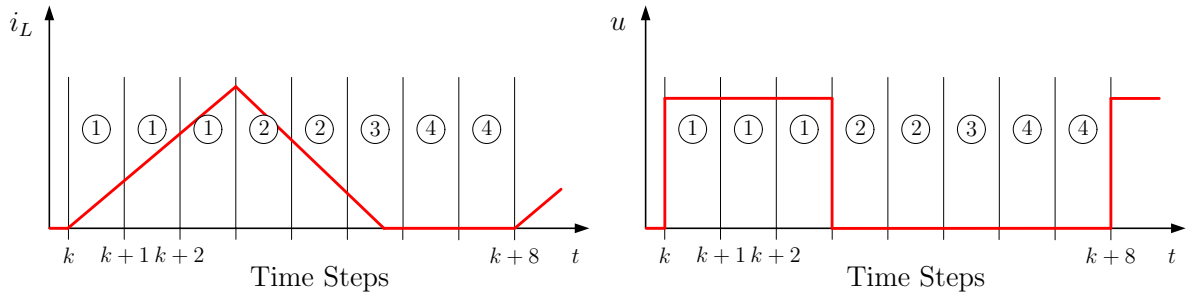
1. The inductor current is positive and the switch is *on* for the whole sampling interval, i.e. $i_L(k) > 0$, $i_L(k+1) > 0$ and $S = 1$.

2. The inductor current is positive and the switch is *off* for the whole sampling interval, i.e. $i_L(k) > 0$, $i_L(k+1) > 0$ and $S = 0$.

3. During the sampling interval the inductor current reaches zero, while the switch is *off*, i.e. $i_L(k) > 0$, $i_L(k+1) = 0$ and $S = 0$.

4. The inductor current is zero and the switch is *off* for the whole sampling interval, i.e. $i_L(k) = i_L(k+1) = 0$ and $S = 0$.

The continuous-time equations of the model as given by (3.5) are discretized using the forward Euler approximation approach, resulting in the following discrete-time model



(a) Inductor current. (b) Switch position.

Figure 3.5: Operation modes used in the mathematical model to describe the boost converter. Depending on the shape of the current four different modes are used.

of the converter

$$x(k+1) = \begin{cases} \boldsymbol{E}_1\boldsymbol{x}(k) + \boldsymbol{F}_1 v_s(k) & \text{Mode “1”} \\ \boldsymbol{E}_2\boldsymbol{x}(k) + \boldsymbol{F}_2 v_s(k) & \text{Mode “2”} \\ \boldsymbol{E}_3\boldsymbol{x}(k) + \boldsymbol{F}_3 v_s(k) & \text{Mode “3”} \\ \boldsymbol{E}_4\boldsymbol{x}(k) & \text{Mode “4”} \end{cases} \tag{3.6a}$$

$$\boldsymbol{y}(k) = \boldsymbol{G}\boldsymbol{x}(k) \tag{3.6b}$$

where the matrices are $\boldsymbol{E}_1 = \boldsymbol{I} + \boldsymbol{\Gamma}_1 T_s$, $\boldsymbol{E}_2 = \boldsymbol{I} + \boldsymbol{\Gamma}_2 T_s$, $\boldsymbol{E}_3 = \frac{1}{T_s}(\tau_1 \boldsymbol{E}_2 + \tau_2 \boldsymbol{E}_4)$, $\boldsymbol{E}_4 = \boldsymbol{I} + \boldsymbol{\Gamma}_3 T_s$, $\boldsymbol{F}_1 = \boldsymbol{\Delta} T_s$, $\boldsymbol{F}_2 = \boldsymbol{F}_1$, $\boldsymbol{F}_3 = \boldsymbol{\Delta}\tau_1$, and $\boldsymbol{G} = \boldsymbol{C}$. Furthermore, τ_1 denotes the time-instant within the sampling interval, when the inductor current reaches zero, i.e. $i_L(k + \tau_1/T_s) = 0$, and $\tau_1 + \tau_2 = T_s$. Finally, $\boldsymbol{I}$ is the identity matrix and T_s is the sampling interval. Note that $\boldsymbol{E}_3$ is derived by averaging over modes “2” and “4”.

The four different operating modes of the converter's mathematical model are illustrated in Fig. 3.6. The transitions from one mode to another are specified by conditions, such as the switch position and the value of the current.

3.3 Optimal Control of DC-DC Boost Converters

In this section, the design of the control scheme is presented. For the dc-dc converter, the main control objective is for the output voltage to accurately track its given reference by appropriately manipulating the switch. This is to be achieved despite changes in the input voltage and load. During transients, the output voltage is to be regulated to its new reference value as fast and with as little overshoot as possible.

In the following two different MPC approaches to the control problem will be presented. In the first approach, the control problem is tackled as a current regulation problem, while in the second, as a voltage regulation.

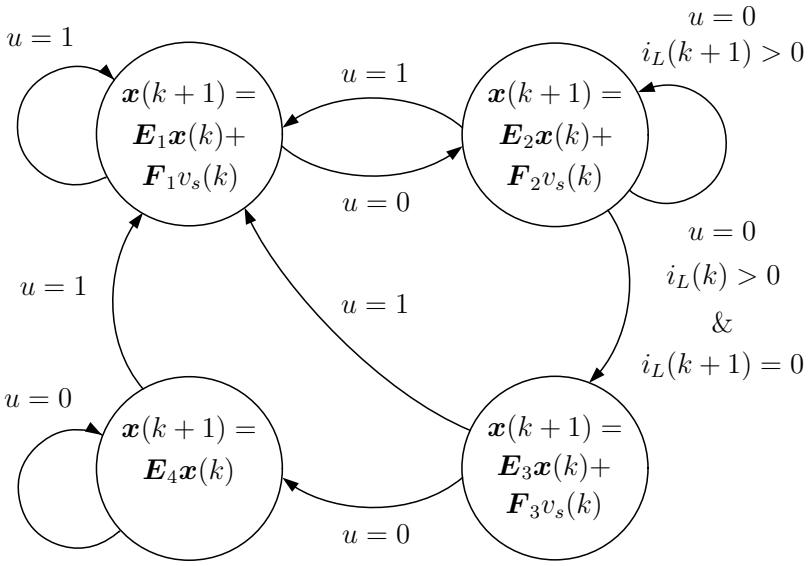


Figure 3.6: Discrete-time mathematical model of the dc-dc converter represented as a discrete-time automaton.

3.3.1 Direct Model Predictive Current Control

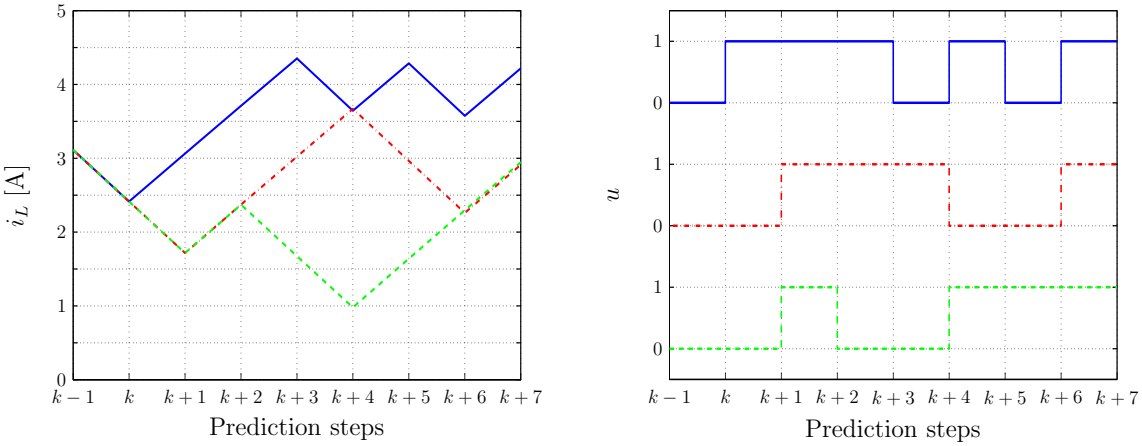
The introduced MPC approach indirectly controls the output voltage by controlling the inductor current (see Fig. 3.1(b)). This is achieved by appropriately manipulating the controllable switch. To derive the optimal sequence of control actions that minimizes a user-defined objective function subject to the plant dynamics, an enumeration technique is used.

3.3.1.1 Objective Function

For the design of the objective function the deviation of the predicted evolution of the variables of concern from the desired behavior, over the horizon N, is taken into consideration. The control input at time-instant kT_s is obtained by minimizing that function over the optimization variable, which is the sequence of switching states over the horizon $\boldsymbol{U}(k) = [\boldsymbol{u}(k)\ \boldsymbol{u}(k+1) \dots \boldsymbol{u}(k+N-1)]^T$. The sequence $\boldsymbol{U}^*$ that minimizes the objective function is the *optimal* solution; the first element of the sequence, denoted as $\boldsymbol{u}^*(k)$, is applied to the converter, the remaining elements are discarded and the procedure is repeated at the successive sampling instant based on new acquired measurements.

An illustrative example of the predicted state—here the inductor current—and the sequence of the control actions, i.e. the switching state, is depicted in Fig. 3.7. Three candidate switching sequences are shown for the prediction horizon $N = 7$. Note that the current that corresponds to time-step k is the measured one, while from $k+1$ to $k+N$ the currents are predicted, assuming the switching sequences shown in Fig. 3.7(b).

Since in the control method introduced here, the control problem is formulated as



(a) Predicted current trajectories (b) Predicted switching sequences

Figure 3.7: Three candidate switching sequences for the prediction horizon $N = 7$.

a current regulation problem, the deviation of the inductor current from its reference defined as

$$i_{L,\mathrm{err}}(k) = i_{L,\mathrm{ref}} - i_L(k)\,, \tag{3.7}$$

is taken into account.

In order to precisely describe the control problem two different objective functions are proposed. In the first approach, the average value of the current error is penalized, while in the second one the rms value of the current error is considered. This allows us to use a shorter prediction horizon.

In the following, the two alternative formulations of the objective function are described.

1. *Average current error*: At time-step k, the average current error over the prediction interval NT_s is given by:

$$i_{L,\mathrm{err},avg}(k) = \frac{1}{NT_s}\int_{kT_s}^{(k+N)T_s} |i_{L,\mathrm{err}}(t|k)|dt\,. \tag{3.8}$$

Exploiting the fact that the current slope changes only at the sampling instants and that in between the sampling instants the slope remains effectively constant[3], the above integral can be rewritten as:

$$i_{L,\mathrm{err},avg}(k) = \frac{1}{N}\sum_{\ell=k}^{k+N-1} |\bar{i}_{L,\mathrm{err}}(\ell|k)| \tag{3.9}$$

[3]Strictly speaking, the current slope is constant only for modes "1", "2" and "4". For mode "3", when the converter transitions from CCM to DCM, the slope is constant for τ_1, while for τ_2 it is zero. However, the error resulting from the approximation given by (3.9) is negligible.

with $\bar{i}_{L,\mathrm{err}}(\ell|k) = \frac{i_{L,\mathrm{err}}(\ell|k)+i_{L,\mathrm{err}}(\ell+1|k)}{2}$.

Based on this, the objective function

$$J_{avg}(k) = \sum_{\ell=k}^{k+N-1} \frac{1}{N}|\bar{i}_{L,\mathrm{err}}(\ell|k)| + \lambda|\Delta \boldsymbol{u}(\ell|k)| \tag{3.10}$$

can be formulated. The second term in (3.10) penalizes the difference between two consecutive switching states

$$\Delta \boldsymbol{u}(k) = \boldsymbol{u}(k) - \boldsymbol{u}(k-1)\,. \tag{3.11}$$

This term is added to decrease the switching frequency and to avoid excessive switching. The weighting factor $\lambda > 0$ sets the trade-off between the inductor current error and the switching frequency. In [31] some guidelines for tuning the weighting factor are given. Furthermore, it should be noted that the switching frequency varies depending on the operating point of the converter. The sampling interval T_s serves as an upper bound on the switching frequency, i.e. $f_{sw} \leq 1/(2T_s)$; regardless of the operating point, the switching frequency cannot be higher than half the sampling frequency. The equality corresponds to the case when $\lambda = 0$, the output voltage is twice the input voltage, i.e. $v_o = 2v_s$, and when the inductor is ideal with $R_L = 0$.

2. *Rms current error*: The rms value of the current error over the prediction interval is equal to

$$i_{L,\mathrm{err},rms}(k) = \sqrt{\frac{1}{NT_s}\int_{kT_s}^{(k+N)T_s} i_{L,\mathrm{err}}(t|k)^2 dt} \tag{3.12}$$

with the current error as given in (3.7). This expression is equivalent to

$$i_{L,\mathrm{err},rms}(k) = \frac{2}{3N}\sum_{\ell=k}^{k+N-1} 2\bar{i}_{L,\mathrm{err}}(\ell|k)^2 - \tilde{i}_{L,\mathrm{err}}(\ell|k) \tag{3.13}$$

with $\tilde{i}_{L,\mathrm{err}}(\ell|k) = \frac{i_{L,\mathrm{err}}(\ell|k)\cdot i_{L,\mathrm{err}}(\ell+1|k)}{2}$.

Based on (3.13) the objective function for the rms current error-based approach is formulated as

$$J_{rms}(k) = \sum_{\ell=k}^{k+N-1} \frac{2}{3N}\Big(2\bar{i}_{L,\mathrm{err}}(\ell|k)^2 - \tilde{i}_{L,\mathrm{err}}(\ell|k)\Big) + \lambda\Delta \boldsymbol{u}(\ell|k)^2\,. \tag{3.14}$$

3.3.1.2 Optimization Problem

Subsequently, for both approaches, an optimization problem is formulated and solved at each sampling instant. This is of the form

$$\begin{aligned} &\text{minimize} && J_\dagger(k) \\ &\text{subject to} && (3.6)\,, \end{aligned} \tag{3.15}$$

where $J_\dagger$ denotes the objective function to be minimized, which is either J_{avg} or J_{rms}, as given by (3.10) and (3.14), respectively.

The optimization problem (3.15) is solved using an enumeration strategy, as explained in Section 2.4. Therefore, all possible combinations of the switching state ($u = 0$ or $u = 1$) over the prediction horizon N are enumerated, yielding the so-called *switching sequences* $\boldsymbol{U}$. There exist 2^N switching sequences. For each switching sequence, the evolution of the variables of concern is calculated using (3.6) and the objective function is evaluated. The switching sequence that results in the minimum cost is chosen as the optimal one, $\boldsymbol{U}^*$. Hence, the control input at time-step k, in both cases, is obtained by minimizing the corresponding objective function, and it is given by

$$\boldsymbol{U}^*(k) = \arg\min J_\dagger(k)\,. \tag{3.16}$$

3.3.1.3 Outer Loop

Since the control problem is formulated as a current regulation problem, an outer loop must be designed (see Fig. 3.1(b)). The reference current for the inner loop is derived from this outer loop based on a feed-forward scheme, using the power balance equation $P_{in} = P_{out}$.

Assuming that the power switches are ideal, the following expression for the desired current results:

$$I_{L,\text{des}} = \frac{V_s}{2R_L} - \sqrt{\left(\frac{V_s}{2R_L}\right)^2 - \frac{V_{o,\text{ref}}^2}{RR_L}} \tag{3.17}$$

In the above equation small-ripple approximation is used [36], i.e. $v_s \approx V_s$ and $v_{o,\text{ref}} \approx V_{o,\text{ref}}$.

In order to further improve the transient response of the output voltage, a term proportional to the voltage error, i.e. $v_{o,\text{ref}} - v_o$, is added to (3.17). Hence, the reference inductor current is given by

$$I_{L,\text{ref}} = I_{L,\text{des}} + h(V_{o,\text{ref}} - v_o)\,, \tag{3.18}$$

with $h \in \mathbb{R}^+$. In (3.18) the small-ripple approximation is used again.

3.3.1.4 Load Variations

As can be seen in Section 3.2, the model of the converter depends on the load. This means that the load has been assumed to be time-invariant and known. In the vast majority of the applications, however, this is not the case; the load typically varies in an unknown way, resulting in a model mismatch and therefore in a steady-state output voltage error. To overcome this, an additional external loop that provides state estimates needs to be designed. Moreover, this loop will adjust the current reference so as to remove the steady-state error between the inductor current and its reference.

Even though a PI-based loop might suffice to meet the two objectives mentioned above, in this work a discrete-time Kalman filter [110] is implemented. Thanks to its integrating nature, the Kalman filter provides offset-free output voltage tracking, while not being operating point dependent.

The model of the converter given by (3.6) is augmented by two integrating disturbance states, i_e and v_e, that model the effect of load variations on the inductor current and the output voltage, respectively. Hence, the Kalman filter estimates the augmented state vector

$$\boldsymbol{x}_a = \begin{bmatrix} i_L & v_o & i_e & v_e \end{bmatrix}^T , \tag{3.19}$$

consisting of the measured state variables, i_L and v_o, and the disturbance states.

The Kalman filter is used to estimate the state vector given by (3.19). Depending on the operating mode of the converter, as shown in Fig. 3.5, four different affine systems result. The respective stochastic discrete-time state equations of the augmented model are

$$\boldsymbol{x}_a(k+1) = \boldsymbol{E}_{za}\boldsymbol{x}_a(k) + \boldsymbol{F}_{za}v_s(k) + \boldsymbol{\xi}(k)\,, \tag{3.20}$$

where $z = \{1, 2, 3, 4\}$ corresponds to the four operating modes of the converter.

The measured state vector is given by

$$\boldsymbol{x}(k) = \begin{bmatrix} i_L(k) \\ v_o(k) \end{bmatrix} = \boldsymbol{G}_a\boldsymbol{x}_a(k) + \boldsymbol{\nu}(k) \tag{3.21}$$

and the matrices are

$$\boldsymbol{E}_{za} = \begin{bmatrix} \boldsymbol{E}_z & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix},\ \boldsymbol{F}_{1a} = \begin{bmatrix} \boldsymbol{F}_1 \\ 0 \\ 0 \end{bmatrix} = \boldsymbol{F}_{2a} = \begin{bmatrix} \boldsymbol{F}_2 \\ 0 \\ 0 \end{bmatrix},\ \boldsymbol{F}_{3a} = \begin{bmatrix} \boldsymbol{F}_3 \\ 0 \\ 0 \end{bmatrix},$$

$$\boldsymbol{F}_{4a} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T,\ \text{and}\ \ \boldsymbol{G}_a = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{I} \end{bmatrix},$$

where, $\boldsymbol{I}$ is the identity matrix of dimension two and $\boldsymbol{0}$ are square zero matrices of dimension two. The variables $\boldsymbol{\xi} \in \mathbb{R}^4$ and $\boldsymbol{\nu} \in \mathbb{R}^2$ denote the process and the measurement

noise, respectively. These terms represent zero-mean, white Gaussian noise sequences with normal probability distributions. Their covariances are given by $E[\boldsymbol{\xi}\boldsymbol{\xi}^T] = \boldsymbol{Q}$ and $E[\boldsymbol{\nu}\boldsymbol{\nu}^T] = \boldsymbol{R}$, and are positive semi-definite and positive definite, respectively.

A switched discrete-time Kalman filter is designed based on the augmented model of the converter. The active mode of the Kalman filter (one out of four) is determined by the switching position and the operating mode of the converter.

Due to the fact that the state-update for each operating mode is different, four Kalman gains K_z need to be calculated. Consequently, the equation for the estimated state $\hat{\boldsymbol{x}}_a(k)$ is

$$\hat{\boldsymbol{x}}_a(k+1) = \boldsymbol{E}_{za}\hat{\boldsymbol{x}}_a(k) + K_z\boldsymbol{G}_a\big(\boldsymbol{x}_a(k) - \hat{\boldsymbol{x}}_a(k)\big) + \boldsymbol{F}_{za}v_s(k)\,. \tag{3.22}$$

The Kalman gains are calculated based on the noise covariance matrices, $\boldsymbol{Q}$ and $\boldsymbol{R}$. These matrices are chosen such that high credibility is assigned to the measurements of the physical states (i_L and v_o), and low credibility to the dynamics of the disturbance states (i_e and v_e). As a result, the Kalman filter provides estimates of the disturbances that can be used to remove their influence from the output voltage and inductor current. The estimated disturbance state $\hat{v}_e$ is used to adjust the output voltage reference $v_{o,\text{ref}}$

$$\tilde{v}_{o,\text{ref}} = v_{o,\text{ref}} - \hat{v}_e\,. \tag{3.23}$$

Hence, in (3.17) and (3.18) the *modified* voltage reference $\tilde{v}_{o,\text{ref}}$ is taken into consideration, instead of the given value $v_{o,\text{ref}}$.

Following the same procedure, the inductor current reference $i_{L,\text{ref}}$ is adjusted using the corresponding estimated disturbance state $\hat{i}_e$, i.e.

$$\tilde{i}_{L,\text{ref}} = i_{L,\text{ref}} - \hat{i}_e\,. \tag{3.24}$$

Moreover, the controller is based on the estimated states $\hat{v}_o$ and $\hat{i}_L$, rather than on the measured ones, v_o and i_L.

3.3.1.5 Control Algorithm

The proposed control technique is summarized in Algorithm 3.1. The function $f_\dagger$ stands for the state-update given by (3.6), and $g_\dagger$ refers to the function that calculates the current error according to (3.9) or (3.13). For the average current error based approach, $p = 1$ is used, whilst for the rms current error based one, $p = 2$ is chosen. In Fig. 3.8 the control diagram of the proposed control strategy including both loops is depicted.

3.3.2 Direct Model Predictive Voltage Control

In this section an MPC voltage-mode scheme for dc-dc boost converters is presented. Thereby, the output voltage is directly controlled by manipulating the controllable switch

Algorithm 3.1 Direct Current-Mode MPC algorithm

function $\boldsymbol{u}^*(k) =$ CURRMPC $(\hat{\boldsymbol{x}}(k), \boldsymbol{u}(k-1))$
  $J_\dagger^*(k) = \infty$; $\boldsymbol{u}^*(k) = \emptyset$; $\boldsymbol{x}(k) = \hat{\boldsymbol{x}}(k)$
  for all $\boldsymbol{U}$ over N **do**
    $J_\dagger = 0$
    for $\ell = k$ to $k+N-1$ **do**
      $\boldsymbol{x}(\ell+1) = f_\dagger(\boldsymbol{x}(\ell), \boldsymbol{u}(\ell))$
      $i_{L,\mathrm{err},\dagger}(\ell) = g_\dagger(\boldsymbol{x}(\ell), \boldsymbol{x}(\ell+1))$
      $\Delta\boldsymbol{u}(\ell) = \boldsymbol{u}(\ell) - \boldsymbol{u}(\ell-1)$
      $J_\dagger = J_\dagger + i_{L,\mathrm{err},\dagger}(\ell) + \lambda|\Delta\boldsymbol{u}(\ell)|^p$
    end for
    if $J_\dagger < J_\dagger^*(k)$ **then**
      $J_\dagger^*(k) = J_\dagger$, $\boldsymbol{u}^*(k) = \boldsymbol{U}(1)$
    end if
  end for
end function

S (Fig. 3.2). To do so, an optimization problem is formulated and solved in real-time. Using an enumeration technique, the user-defined objective function is minimized subject to the converter dynamics.

3.3.2.1 Objective Function

Since the control problem is formulated as a voltage regulation problem (see Fig. 3.1(a)) the objective function should consist of a voltage term. Thus, the objective function is chosen as

$$J(k) = \sum_{\ell=k}^{k+N-1} \Big(|v_{o,\mathrm{err}}(\ell+1|k)| + \lambda|\Delta\boldsymbol{u}(\ell|k)| \Big) \tag{3.25}$$

which penalizes the absolute values of the variables of concern over the prediction horizon N, which is of finite length. The first term penalizes the absolute value of the output voltage error

$$v_{o,\mathrm{err}}(k) = v_{o,\mathrm{ref}} - v_o(k)\,. \tag{3.26}$$

Furthermore, in order to decrease the switching frequency and to avoid excessive switching, the difference between two consecutive switching states are penalized (as in (3.10) and (3.14)), i.e.

$$\Delta\boldsymbol{u}(k) = \boldsymbol{u}(k) - \boldsymbol{u}(k-1)\,. \tag{3.27}$$

In (3.25) the weighting factor $\lambda > 0$ sets the trade-off between output voltage error and switching frequency, f_{sw}. Once again, the sampling interval T_s implicitly imposes an

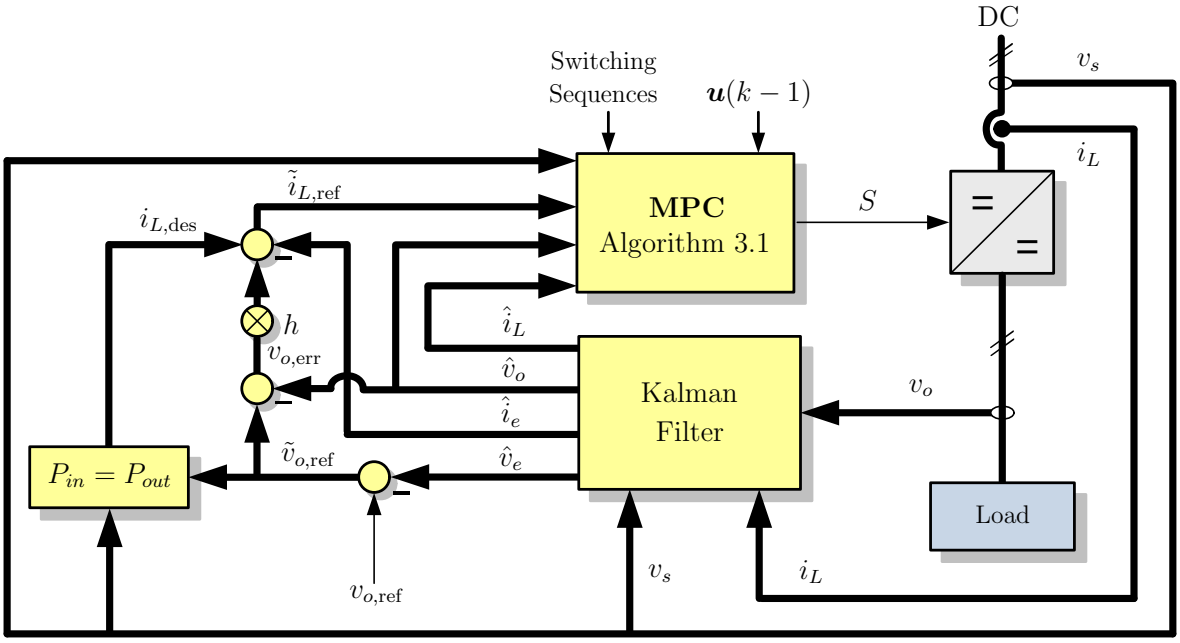


Figure 3.8: Control diagram of the direct model predictive current control scheme.

upper bound on the switching frequency, i.e. $f_{sw} \leq 1/(2T_s)$, as explained in Section 3.3.1.

3.3.2.2 Optimization Problem

The optimization problem underlying MPC at time-step k amounts to minimizing the objective function (3.25) subject to the converter model dynamics

$$\begin{aligned} &\text{minimize} && J(k) \\ &\text{subject to} && (3.6)\,. \end{aligned} \tag{3.28}$$

As in the current optimization problem (3.15), the optimization variable is the sequence of switching states over the horizon, which is $\boldsymbol{U}(k) = [\boldsymbol{u}(k)\ \boldsymbol{u}(k+1) \dots \boldsymbol{u}(k+N-1)]^T$. Minimizing (3.28) yields the optimal switching sequence $\boldsymbol{U}^*(k)$

$$\boldsymbol{U}^*(k) = \arg\min J(k)\,. \tag{3.29}$$

Out of this sequence, the first element $\boldsymbol{u}^*(k)$ is applied to the converter. The procedure is repeated at $k+1$, based on new measurements acquired at the following sampling instance.

The optimization problem (3.28) is a mixed-integer nonlinear optimization problem (like problem (3.15)). This is due to the fact that the mathematical model of the converter given by (3.6a) for modes "1" and "2" is affine, and for mode "4" is linear, while the expression for mode "3" is nonlinear. Therefore minimizing (3.28) is a challenging task.

A straightforward alternative is to solve it using enumeration (see Section 2.4). Therefore, the procedure followed involves the following three steps. First, by considering all

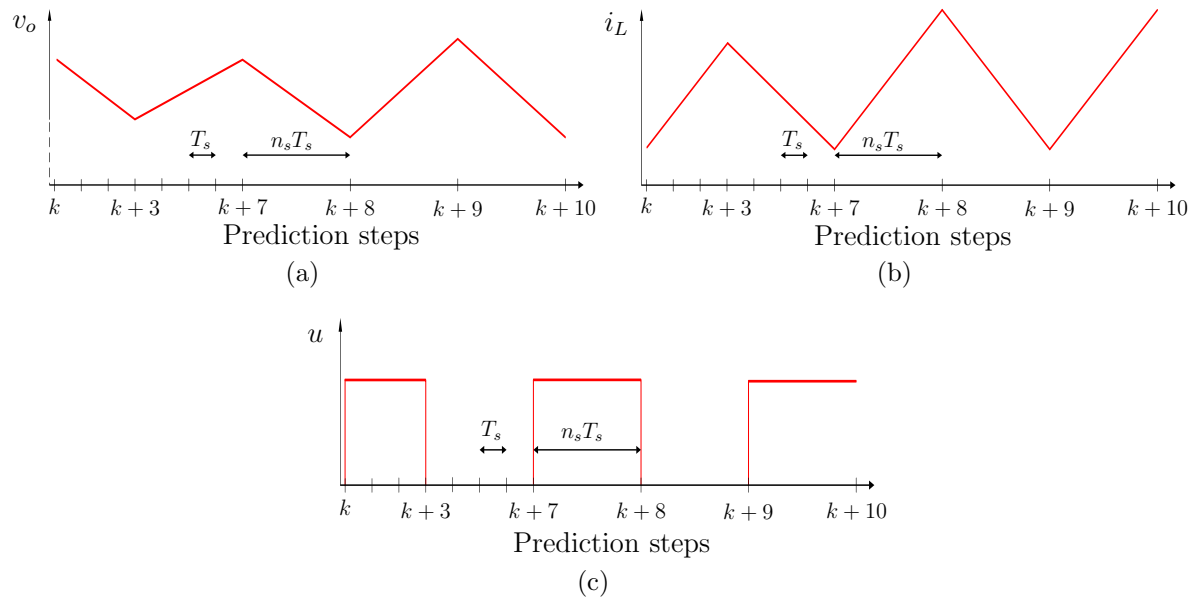


Figure 3.9: Prediction horizon with move blocking: a) output voltage, b) inductor current, and c) control input. The prediction horizon has $N = 10$ time-steps, but the prediction interval is of length $19T_s$, since $n_s = 4$ is used for the last $N_2 = 3$ steps.

possible combinations of the switching states ($u = 0$ or $u = 1$) over the prediction horizon, the set of admissible switching sequences is assembled. For each of the 2^N sequences, the corresponding output voltage trajectory is predicted and the objective function is evaluated. The optimal switching sequence is obtained by choosing the one with the smallest associated cost.

3.3.2.3 Move Blocking Scheme

A fundamental difficulty associated with boost converters arises when controlling their output voltage without an intermediate current control loop, since the output voltage exhibits a nonminimum phase behavior with respect to the switching action. For example, when increasing the output voltage, the duty cycle of switch S has to be ramped up, but initially the output voltage drops before increasing. This implies that the sign of the gain (from the duty cycle to the output voltage) is not always positive.

To overcome this obstacle and to ensure closed-loop stability, a sufficiently long prediction interval NT_s is required, so that the controller can "see" beyond the initial voltage drop when contemplating to increase the duty cycle. On the one hand, increasing N leads to an exponential increase in the number of switching sequences to be considered and thus dramatically increases the number of calculations needed. On the other hand, long sampling intervals T_s reduce the resolution of the possible switching instants, since switching can only be performed at the sampling instants.

A long prediction interval NT_s with a small N and a small T_s can be achieved by

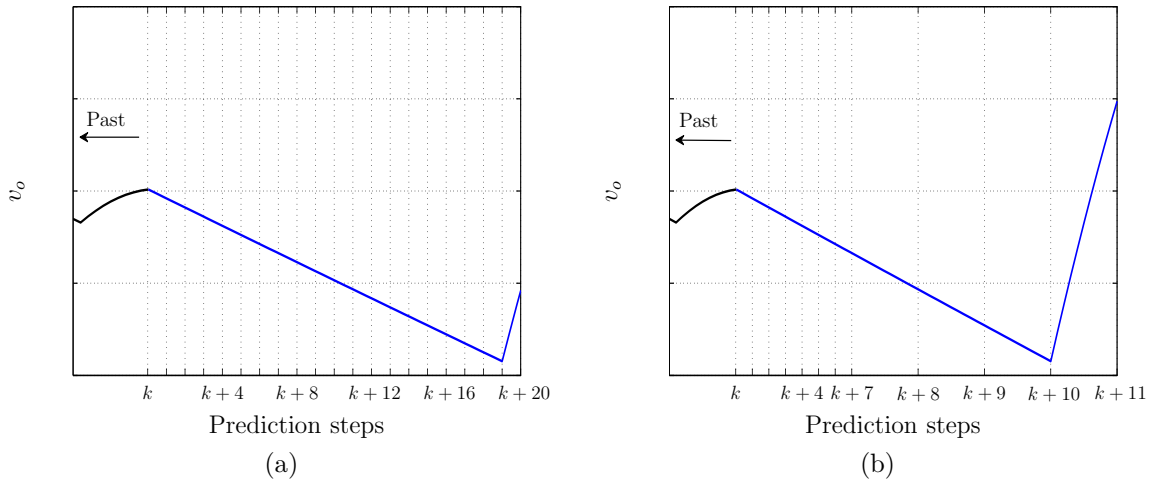


Figure 3.10: Effect of the move blocking scheme. In (a), without move blocking, a prediction horizon of $N = 20$ steps of equal time-intervals is needed. In (b), with the move blocking strategy employed, an $N = 11$ prediction horizon is sufficient to achieve the same closed-loop result ($N_1 = 7$, $N_2 = 4$, and $n_s = 4$, total length $23T_s$).

employing a *move blocking* technique [20]. For the first steps in the prediction horizon, the prediction model is sampled with T_s, while for steps far in the future, the model is sampled more coarsely with a multiple of T_s, i.e. n_sT_s, with $n_s \in \mathbb{N}^+$ [46]. As a result, different sampling intervals are used within the prediction horizon, as illustrated in Fig. 3.9. We use N_1 to denote the number of prediction steps in the first part of the horizon, which are sampled with T_s. Accordingly, N_2 refers to the number of steps in the last part of the horizon, sampled with n_sT_s. The total number of time-steps in the horizon is $N = N_1 + N_2$.

An illustrative example of the effectiveness of the move blocking strategy is depicted in Fig 3.10. Assume that at time instant kT_s the output voltage reference increases in a stepwise manner and the output voltage is to follow that change. However, as mentioned above, because of the nonminimum phase nature of the system, the output voltage initially tends to decrease. In order to ensure that MPC is able to predict the final voltage increase and will thus pick the corresponding switching sequence that achieves this, in this example, a prediction interval of twenty time-steps is required, i.e. $NT_s = 20$.

By employing the move blocking scheme, the eleven-step horizon $N = 11$, with $N_1 = 7$, $N_2 = 4$, and $n_s = 4$ suffices, resulting in a prediction interval of a 23 time-steps. In this way, the computational cost is significantly reduced. Without move blocking, the number of switching sequences to be examined is $2^{20} = 1048576$, and the state evolution has to be predicted for 20 steps into the future. In contrast to this, when using the move blocking scheme, the total number of sequences is $2^{11} = 2048$, and the evolution of the state needs to be calculated only for 11 steps. As a result, the computations required are

decreased by three orders of magnitude, or 99.9%.

It is important to point out that a high timing resolution is required only around the current time-step and the very near future. Further ahead, a rough timing resolution suffices, due to the receding horizon policy. The coarse plan of the second part of the prediction horizon is step by step shifted towards the beginning of the prediction horizon and simultaneously refined.

3.3.2.4 Load Variations

In order to address time-varying and unknown loads, a Kalman filter is added, as the one presented in Section 3.3.1.4. The Kalman filter estimates the converter states and provides offset-free tracking of the output voltage due to its integrating action, despite changes in the load. In that way the robustness of the controller is ensured even when the converter operates under nonnominal conditions.

Therefore, this additional loop is employed to provide state estimates to the previously derived optimal controller, where the load was assumed to be known and constant. The output voltage reference will be adjusted so as to compensate for the deviation of the output voltage from its actual reference.

Two integrating disturbance states, i_e and v_e, are introduced in order to model the effect of the load variations on the inductor current and output voltage, respectively. The measured state variables, i_L and v_o, together with the disturbance state variables form the augmented state vector given by (3.19), which is repeated here for the convenience of the reader

$$\boldsymbol{x}_a = \begin{bmatrix} i_L & v_o & i_e & v_e \end{bmatrix}^T . \tag{3.30}$$

consisting of the measured state variables, i_L and v_o, and the disturbance states.

Hence, the stochastic discrete-time state equation of the augmented model is given by (3.20), and repeated here

$$\boldsymbol{x}_a(k+1) = \boldsymbol{E}_{za}\boldsymbol{x}_a(k) + \boldsymbol{F}_{za}v_s(k) + \boldsymbol{\xi}(k)\,, \tag{3.31}$$

with $z = \{1, 2, 3, 4\}$, corresponding to the four modes of operation.

The measurement equation (3.21) is

$$\boldsymbol{x}(k) = \begin{bmatrix} i_L(k) \\ v_o(k) \end{bmatrix} = \boldsymbol{G}_a\boldsymbol{x}_a(k) + \boldsymbol{\nu}(k)\,. \tag{3.32}$$

where the matrices $\boldsymbol{E}_{za}$, $\boldsymbol{F}_{za}$, and $\boldsymbol{G}_a$ are the same as in Section 3.3.1.4.

The process noise is denoted by $\boldsymbol{\xi} \in \mathbb{R}^4$ and the measurement noise by $\boldsymbol{\nu} \in \mathbb{R}^2$. Both of the noise disturbances represent zero-mean, white Gaussian noise sequences with normal probability distributions. The process noise covariance matrix is positive semi-definite

and it is given by $E[\boldsymbol{\xi}\boldsymbol{\xi}^T] = \boldsymbol{Q}$. The measurement noise covariance matrix is given by $E[\boldsymbol{\nu}\boldsymbol{\nu}^T] = \boldsymbol{R}$, and it is positive definite.

The switched discrete-time Kalman filter implemented is given by (3.22), i.e.

$$\hat{\boldsymbol{x}}_a(k+1) = \boldsymbol{E}_{za}\hat{\boldsymbol{x}}_a(k) + K_z\boldsymbol{G}_a\big(\boldsymbol{x}_a(k) - \hat{\boldsymbol{x}}_a(k)\big) + \boldsymbol{F}_{za}v_s(k)\,. \tag{3.33}$$

where K_z is the Kalman gain, i.e. four unique Kalman gains need to be calculated, one for each operating mode. Moreover, the noise covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ are chosen such that high credibility is assigned to the measurements of the physical states (i_L and v_o), whilst low credibility is assigned to the dynamics of the disturbance states (i_e and v_e). The Kalman gains are calculated based on these matrices. The estimated disturbances, provided by the resulting filter, can be used to remove their influence from the output voltage. Hence, the disturbance state $\hat{v}_e$ is used to adjust the output voltage reference $v_{o,\text{ref}}$

$$\tilde{v}_{o,\text{ref}} = v_{o,\text{ref}} - \hat{v}_e\,. \tag{3.34}$$

Thereby, the estimated states, $\hat{i}_L$ and $\hat{v}_o$, are used as inputs to the controller, instead of the measured states, i_L and v_o.

Algorithm 3.2 Direct Voltage-Mode MPC algorithm

```
function u*(k) = VoltMPC (x̂(k), u(k−1))
    J*(k) = ∞; u*(k) = ∅; x(k) = x̂(k)
    for all U over N do
        J = 0
        for ℓ = k to k + N − 1 do
            if ℓ < k + N1 then
                x(ℓ + 1) = f1(x(ℓ), u(ℓ))
            else
                x(ℓ + 1) = f2(x(ℓ), u(ℓ))
            end if
            v_o,err(ℓ + 1) = ṽ_o,ref − v_o(ℓ + 1)
            Δu(ℓ) = u(ℓ) − u(ℓ − 1)
            J = J + |v_o,err(ℓ + 1)| + λ|Δu(ℓ)|
        end for
        if J < J*(k) then
            J*(k) = J, u*(k) = U(1)
        end if
    end for
end function
```

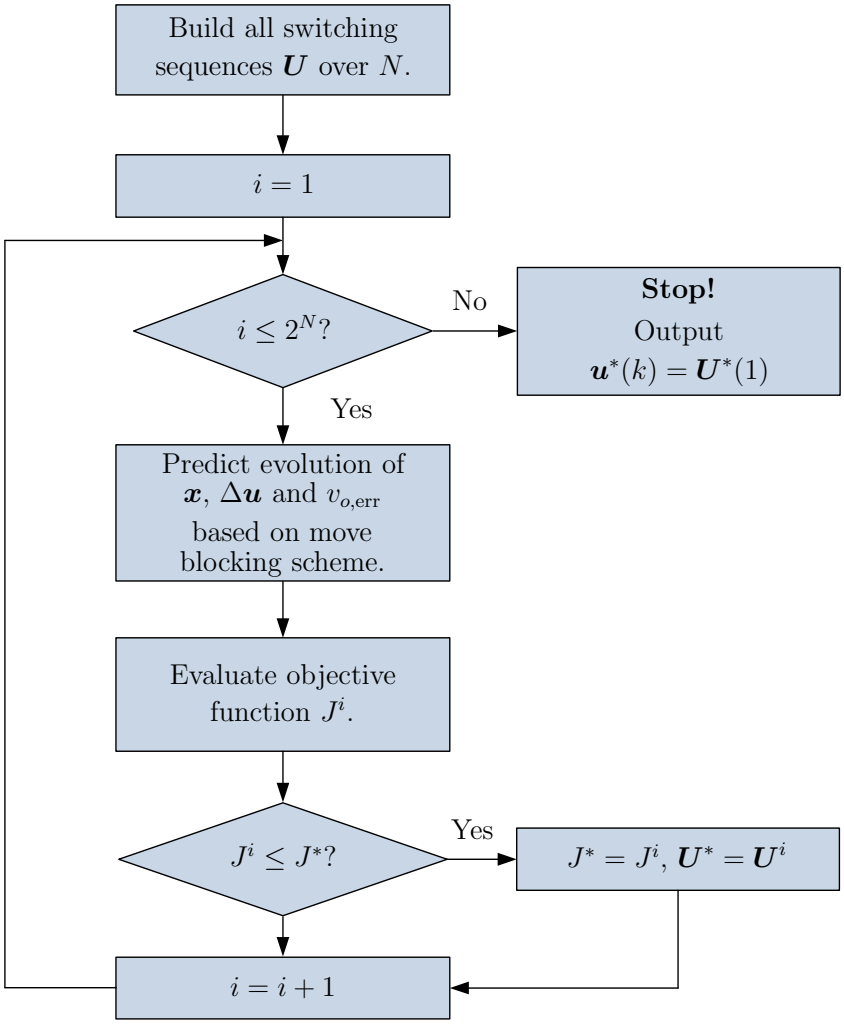


Figure 3.11: Flowchart of the of the direct model predictive voltage control algorithm.

3.3.2.5 Control Algorithm

The proposed control concept is summarized in Algorithm 3.2. The function f stands for the state-update given by (3.6), with the subscripts 1 and 2 corresponding to the sampling interval being used, i.e. T_s and $n_s T_s$, respectively. Fig. 3.11 depicts the flowchart of the introduced MPC algorithm, while the block diagram of the entire control scheme is shown in Fig. 3.12.

3.4 Simulation Results

3.4.1 Current-Mode MPC

In this section simulation results are presented demonstrating the dynamical performance of the proposed model predictive current controller. The simulations focus on the new MPC strategy for the current loop and its dynamical properties; at this point the behavior

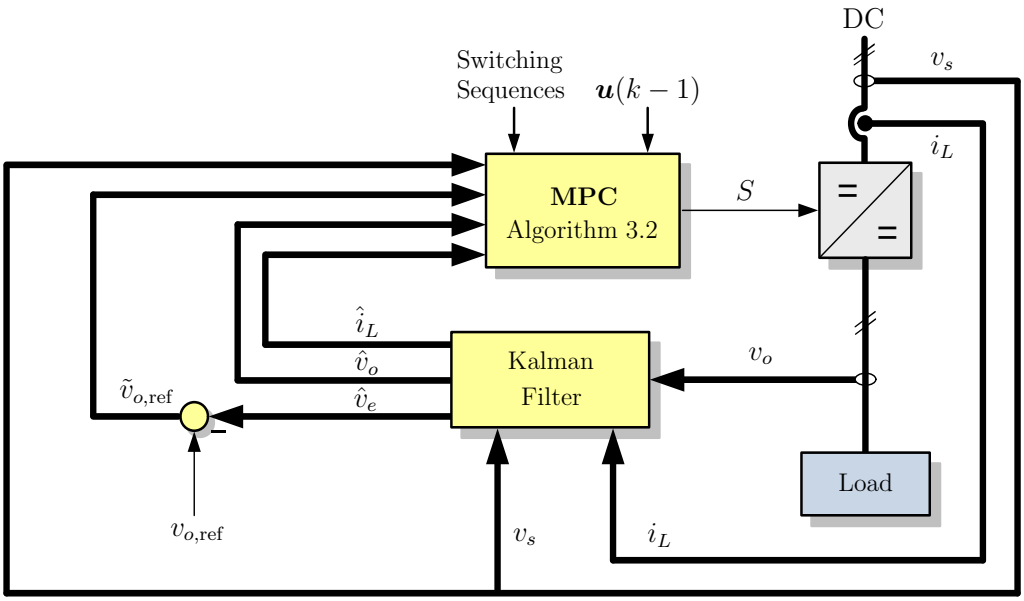


Figure 3.12: Block diagram of the of the direct model predictive voltage control scheme and Kalman filter.

of the whole system is not presented to not obstruct the dynamical analysis. Thus, for both approaches the same scenario is examined, namely a step-down change in the inductor current reference. The behavior of the converter in both CCM and DCM is examined.

The circuit parameters are $L = 150\,\mu\text{H}$, $R_L = 0.2\,\Omega$ and $C_o = 220\,\mu\text{F}$. The load resistance is assumed to be known and constant for all operating points; it is equal to $R = 73\,\Omega$. Initially, the input voltage is $v_s = 20\,\text{V}$, while the output reference voltage is set equal to $v_{o,\text{ref}} = 53.5\,\text{V}$, corresponding to the reference inductor current $i_{L,\text{ref}} = 2\,\text{A}$.

Regarding the objective function, the weighting factor is tuned in such a way that the switching frequency in both approaches is approximately the same, i.e. $\lambda = 0.3$ for the first approach and $\lambda = 0.6$ for the second. The prediction horizon is $N = 5$, and the sampling interval is $T_s = 2.5\,\mu\text{s}$.

The converter initially operates under nominal conditions. At time $t = 0.1\,\text{ms}$, a change to the inductor current reference from $i_{L,\text{ref}} = 2\,\text{A}$ to $i_{L,\text{ref}} = 0.7\,\text{A}$ occurs. As can be seen in Fig. 3.13, for both approaches, the inductor current reaches very quickly the new desired level. The switching frequency is about $f_{sw} \approx 45\,\text{kHz}$. Since the operating points and the corresponding switching frequencies are the same in both approaches, the current ripples observed are identical.

The main difference between the two proposed approaches can be observed in Fig. 3.14, which relates to the converter operating under nominal and steady-state conditions. The impact of varying the weighting factor λ is investigated. The corresponding output volt-

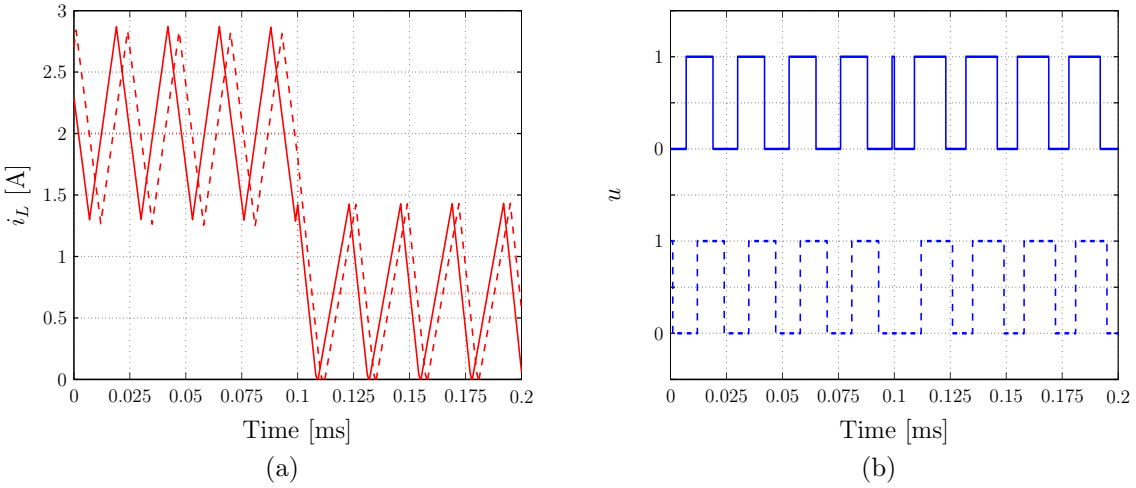


Figure 3.13: Simulation results for the step-down change scenario: a) inductor current for the first (solid line) and the second (dashed line) approach, and inductor current reference (dotted line), b) pulses for the first (solid line) and the second (dashed line) approach.

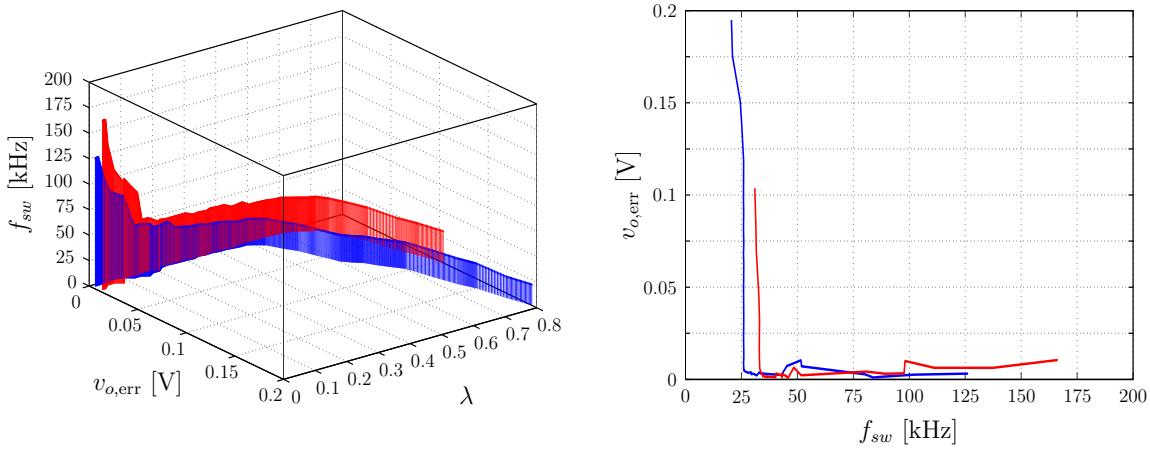
age error, given by

$$v_{o,\text{err}} = \sqrt{\frac{1}{N}\Big(\sum_{k=1}^{N} v_{o,\text{ref}} - v_o(k)\Big)^2}, \tag{3.35}$$

and the switching frequency f_{sw} are depicted. As can be seen, the average current error-based approach results in a lower switching frequency with zero tracking error, which means that lower switching losses can be achieved with this approach. On the other hand, the rms current error-based approach leads to higher switching frequencies, when λ is very small, due to the quadratic penalty. Such high switching frequencies tend to result in even faster transient responses.

This can be seen in Figs. 3.15 and 3.16. When the weighting factor is tuned to be the same in both approaches, i.e. $\lambda = 0.3$ then the dynamical behavior of the system differs. In Fig. 3.15 the response of the controller in a step-up change in the current reference is depicted. At time $t = 0.1\,\text{ms}$, a change to the inductor current reference from $i_{L,\text{ref}} = 2\,\text{A}$ to $i_{L,\text{ref}} = 3\,\text{A}$ occurs. As can be seen in Fig. 3.15, the inductor current very quickly reaches the new desired level, in both approaches. However, as mentioned above, due to the quadratic penalty used in the second approach (rms-based approach) the deviation of the current from its reference is penalized more heavily resulting in a smaller ripple, thus in a higher switching frequency. Because of these reasons the current in the second approach regulates to its reference slightly faster.

Furthermore, a step-down variation in the current reference is investigated. At time $t = 0.1\,\text{ms}$, the reference value changes from $i_{L,\text{ref}} = 3\,\text{A}$ to $i_{L,\text{ref}} = 1\,\text{A}$. The response of the converter is depicted in Fig. 3.16. In both approaches the current decreases very fast



(a) The output voltage error $v_{o,\mathrm{err}}$ and the corresponding switching frequency f_{sw} versus the weighting factor λ.

(b) The output voltage error $v_{o,\mathrm{err}}$ versus the switching frequency f_{sw}.

Figure 3.14: Effect of the weighting factor λ on the output voltage error $v_{o,\mathrm{err}}$ and the switching frequency f_{sw} for the average current error-based (blue) and the rms current error-based (red) approach; the converter operates under nominal conditions.

to its new desired level. The behavior of the controller for both approaches is very similar, and the same observations are made, i.e. the current in the second approach settles to its reference faster. Finally, as can be seen, for the average-based approach, because of the high ripple current the converter operates in DCM, since the current reaches zero for an amount of time.

3.4.2 Voltage-Mode MPC

In this section simulation results are presented to demonstrate the performance of the proposed voltage-mode controller under several operating conditions. Specifically, the closed-loop converter behavior is examined in both CCM and DCM. The dynamic performance is investigated during start-up. Moreover, the responses of the output voltage to step changes in the commanded voltage reference, the input voltage and the load are illustrated.

The circuit parameters are $L = 450\,\mu\mathrm{H}$, $R_L = 0.3\,\Omega$ and $C_o = 220\,\mu\mathrm{F}$. The nominal load resistance is $R = 73\,\Omega$. If not otherwise stated, the input voltage is $v_s = 10\,\mathrm{V}$ and the reference of the output voltage is $v_{o,\mathrm{ref}} = 15\,\mathrm{V}$.

The weight in the objective function is $\lambda = 0.1$, the prediction horizon is $N = 14$ and the sampling interval is $T_s = 2.5\,\mu\mathrm{s}$. A move blocking scheme is used with $N_1 = 8$, $N_2 = 6$ and $n_s = 4$, i.e. the sampling interval for each of the last six steps in the prediction interval is $T_s = 10\,\mu\mathrm{s}$. Note that the length of the prediction horizon in time should be as long as possible. A horizon of about $80\,\mu\mathrm{s}$ is sufficient. The first part of the prediction horizon

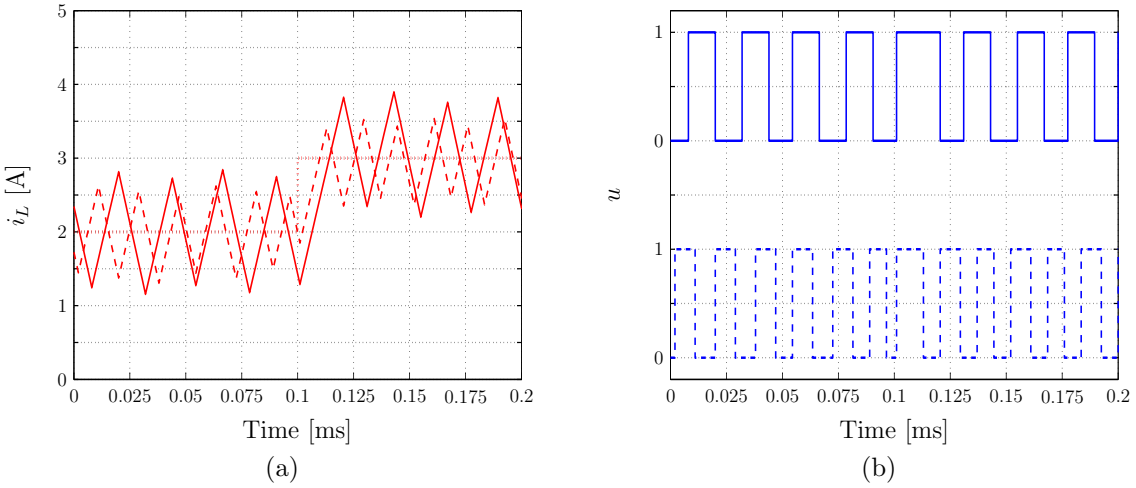


Figure 3.15: Simulation results for the step-up change scenario: a) inductor current for the first (solid line) and the second (dashed line) approach, and inductor current reference (dotted line), b) pulses for the first (solid line) and the second (dashed line) approach.

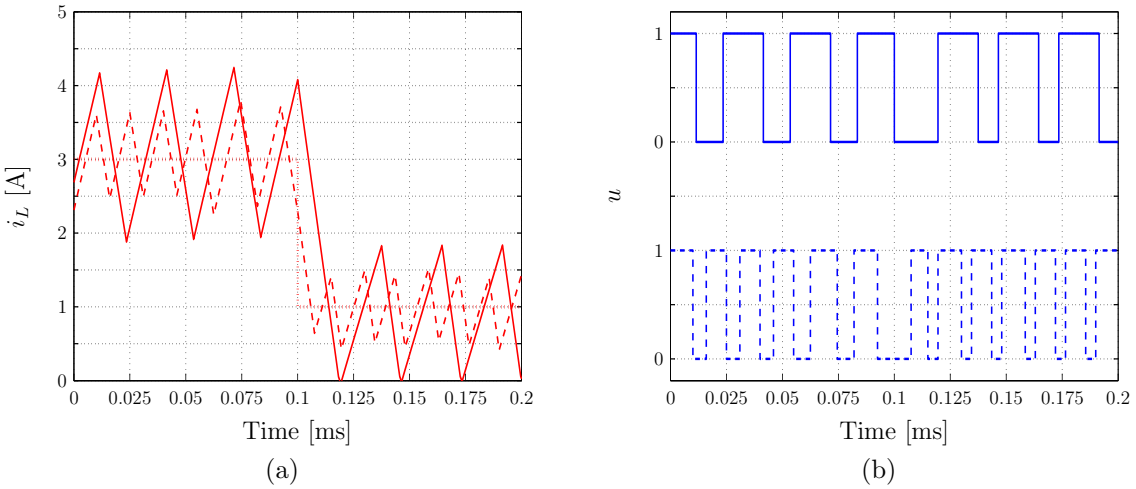


Figure 3.16: Simulation results for the step-down change scenario: a) inductor current for the first (solid line) and the second (dashed line) approach, and inductor current reference (dotted line), b) pulses for the first (solid line) and the second (dashed line) approach.

should be finely sampled, since switching is possible only at the sampling instants. As such, the sampling interval T_s should be as small as possible. The number of steps in the prediction horizon $N = N_1 + N_2$ determines the computational complexity. To ensure that the control law can be computed within T_s, N should be relatively small, leading to the choice made above. Finally, the covariance matrices of the Kalman filter are chosen

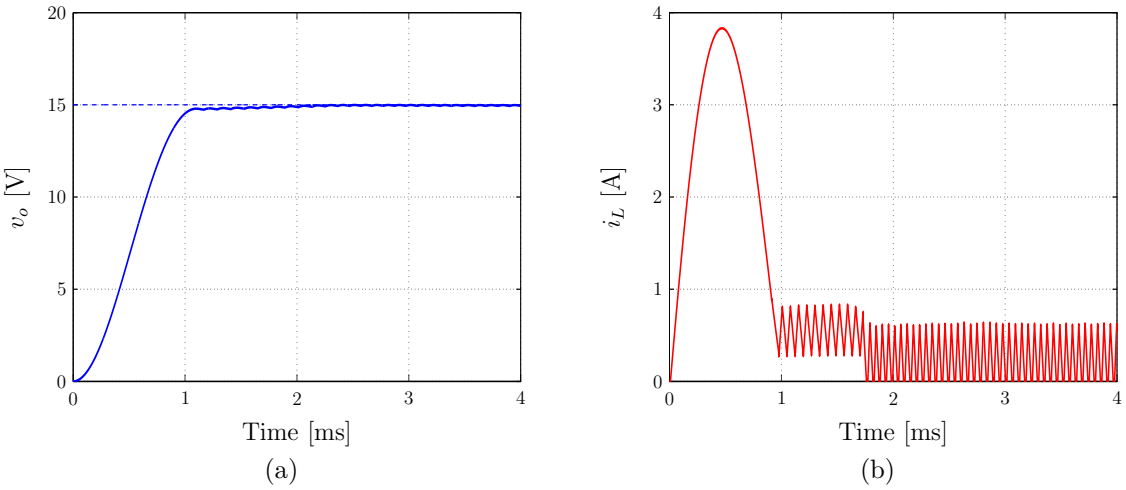


Figure 3.17: Simulation results for nominal start-up: a) output voltage (solid line) and output voltage reference (dashed line), b) inductor current.

as

$$\boldsymbol{Q} = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 50 \end{bmatrix}, \quad \boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

3.4.2.1 Nominal Start-Up

The first case to be examined is that of the start-up behavior under nominal conditions. As can be seen in Fig. 3.17, the inductor current is very quickly increased until the capacitor is charged to the desired voltage level. The output voltage reaches its reference value in about $t \approx 1.8\,\text{ms}$, without any noticeable overshoot. Subsequently, the converter operates in DCM with the inductor current reaching zero.

3.4.2.2 Step Changes in the Output Reference Voltage

Next, step changes in the reference of the output voltage are considered. First, a step-up change in the output reference voltage is examined: at time $t = 2\,\text{ms}$ the reference is doubled from $v_{o,\text{ref}} = 15\,\text{V}$ to $v_{o,\text{ref}} = 30\,\text{V}$. As can be seen in Fig. 3.18, the controller increases the current temporarily in order to quickly ramp up the output voltage. Note that this favorable choice is made by the controller thanks to its long prediction horizon and *despite* the nonminimum phase behavior of the converter. Once the output voltage has reached its reference, the inductor current is decreased to the level that corresponds to the steady-state power balance. The controller exhibits an excellent behavior during the transient, reaching the new output voltage in about $t \approx 1.8\,\text{ms}$, without any overshoot.

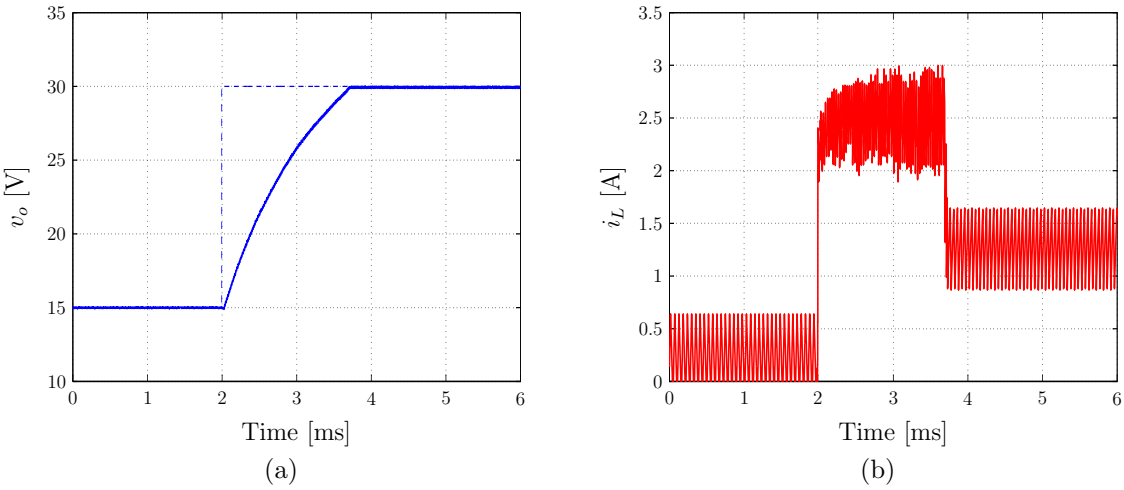


Figure 3.18: Simulation results for a step-up change in the output voltage reference: a) output voltage (solid line) and output voltage reference (dashed line), b) inductor current.

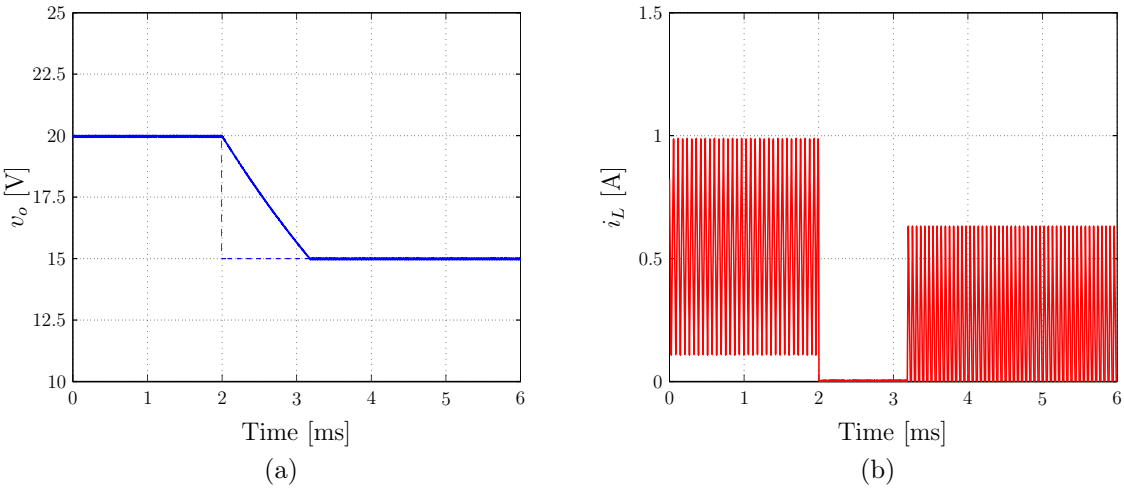


Figure 3.19: Simulation results for a step-down change in the output voltage reference: a) output voltage (solid line) and output voltage reference (dashed line), b) inductor current.

Furthermore, the behavior of the controller is tested under a step-down change in the output reference voltage. At time $t = 2\,\text{ms}$, the output voltage reference changes from $v_{o,\text{ref}} = 20\,\text{V}$ to $v_{o,\text{ref}} = 15\,\text{V}$; the segment of interest is depicted in Fig. 3.19. Since the proposed MPC strategy is formulated as a voltage-mode controller effort is put into decreasing the voltage to its new desired level as quickly as possible. To do so, the controllable switch is turned off, the current instantaneously reaches zero, and the capacitor discharges through the load until it reaches its new demanded value in about $t \approx 1.2\,\text{ms}$.

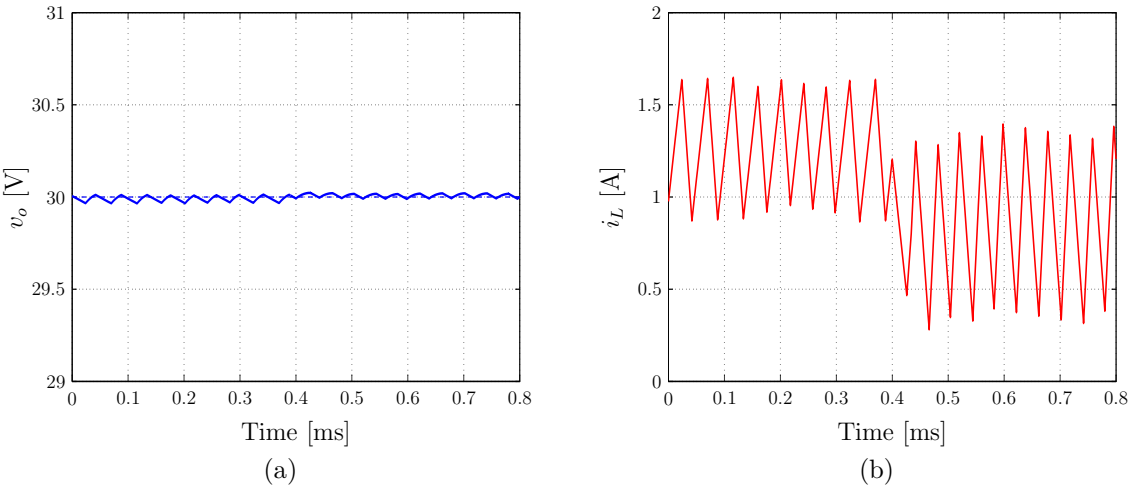


Figure 3.20: Simulation results for a step-up change in the input voltage: a) output voltage (solid line) and output voltage reference (dashed line), b) inductor current.

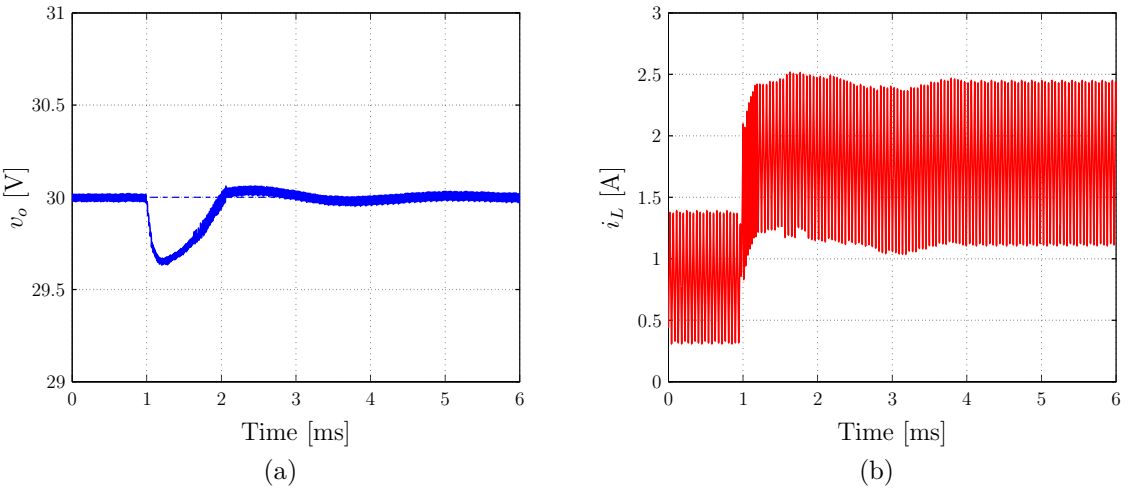


Figure 3.21: Simulation results for a step-down change in the load: a) output voltage (solid line) and output voltage reference (dashed line), b) inductor current.

3.4.2.3 Step Change in the Input Voltage

Operating at the steady-state operating point corresponding to $v_{o,\text{ref}} = 30\,\text{V}$, the input voltage is changed in a step-wise fashion. At time $t = 0.4\,\text{ms}$ the input voltage is increased from $v_s = 10\,\text{V}$ to $v_s = 15\,\text{V}$. The transient response of the converter is depicted in Fig. 3.20. The output voltage remains practically unaffected, with no undershoot observed, while the controller settles very quickly at the new steady-state operating point.

3.4.2.4 Load Step Change

The last case examined is that of a drop in the load resistance. As can be seen in Fig. 3.21, a step-down change in the load from $R = 73\,\Omega$ to $R = 36.5\,\Omega$ occurs at $t = 1\,\text{ms}$ (the input voltage is $v_s = 15\,\text{V}$, and the output voltage reference is $v_{o,\text{ref}} = 30\,\text{V}$). The Kalman filter adjusts the output voltage reference to its new value so as to avoid any steady-state tracking error. This can be observed in Fig. 3.21(a); after the converter has settled at the new operating point, the output voltage accurately follows its reference.

3.5 Experimental Results

To further investigate the potential advantages of the proposed algorithms (the direct model predictive current and voltage controllers), the controllers were implemented on a dSpace DS1104 real-time system. A boost converter was built using an IRF620 MOSFET and a MUR840 diode as active and passive switches, respectively. The physical values of the circuit parameters are $L = 450\,\mu\text{H}$, $R_L = 0.3\,\Omega$ and $C_o = 220\,\mu\text{F}$. The nominal conditions refer to an input voltage of $v_s = 10\,\text{V}$ and a load resistance of $R = 73\,\Omega$. If not otherwise stated, the output voltage reference is $v_{o,\text{ref}} = 15\,\text{V}$. Hall effect transducers were used to acquire the voltage and inductor current measurements.

3.5.1 Current-Mode MPC

The proposed current-mode MPC strategy is executed every $T_s = 15\,\mu\text{s}$ and a prediction horizon of three steps is used ($N = 3$). The weighting factor in the objective function is set to $\lambda = 0.4$. Depending on the tuning of λ, both control approaches yield similar results, as shown in Section 3.4.1. Therefore, it suffices to present the dynamic behavior of only one methodology. This section focuses on the average current error-based approach. Regarding the Kalman filter, the covariance matrices are the same as those in Section 3.4.2, i.e.

$$\boldsymbol{Q} = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 50 \end{bmatrix}, \quad \boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

3.5.1.1 Nominal Start-Up

First, the dynamic behavior of the converter during start-up and nominal conditions is investigated. As can be seen in Fig. 3.22(b), the inductor current quickly increases in order to charge the capacitor to the desired voltage level. The output voltage reaches its

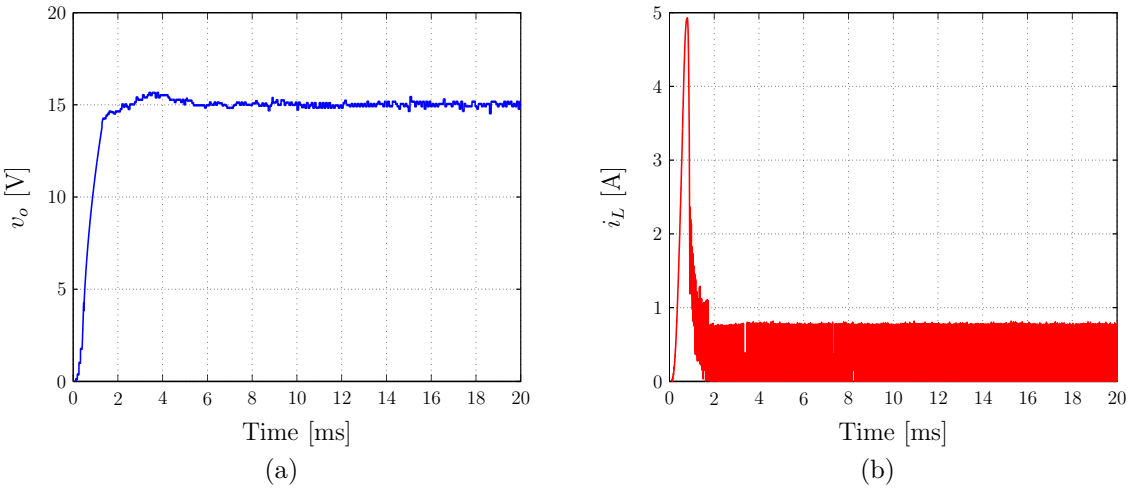


Figure 3.22: Experimental results for nominal start-up: a) output voltage, and b) inductor current.

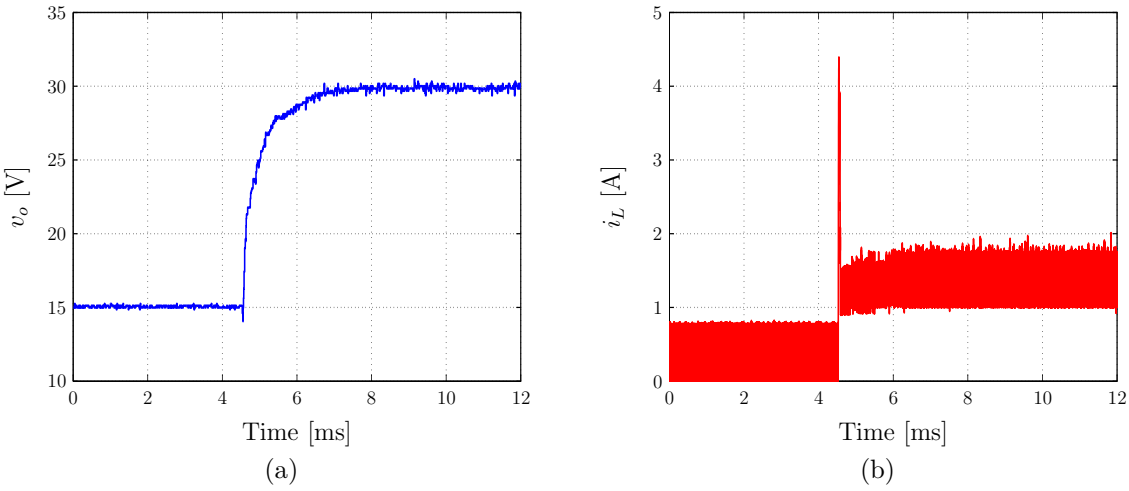


Figure 3.23: Experimental results for a step-up change in the output voltage reference: a) output voltage, and b) inductor current.

reference in $t \approx 3\,\text{ms}$ with a small overshoot, see Fig. 3.22(a). After the transient, the inductor current reaches its nominal value and the converter operates in DCM.

3.5.1.2 Step Change in the Output Reference Voltage

Next, a step-up change in the reference of the output voltage is considered. At time instant $t \approx 4.5\,\text{ms}$ the output voltage reference steps up from its initial value, i.e. from $v_{o,\text{ref}} = 15\,\text{V}$ to $v_{o,\text{ref}} = 30\,\text{V}$, see Fig. 3.23. As previously, the inductor current rapidly increases (Fig. 3.23(b)) so as to charge the capacitor to the new desired level. Initially, the output voltage briefly decreases due to the nonminimum phase characteristic of the system, before it increases, see Fig. 3.23(a), reaching its reference value without an overshoot

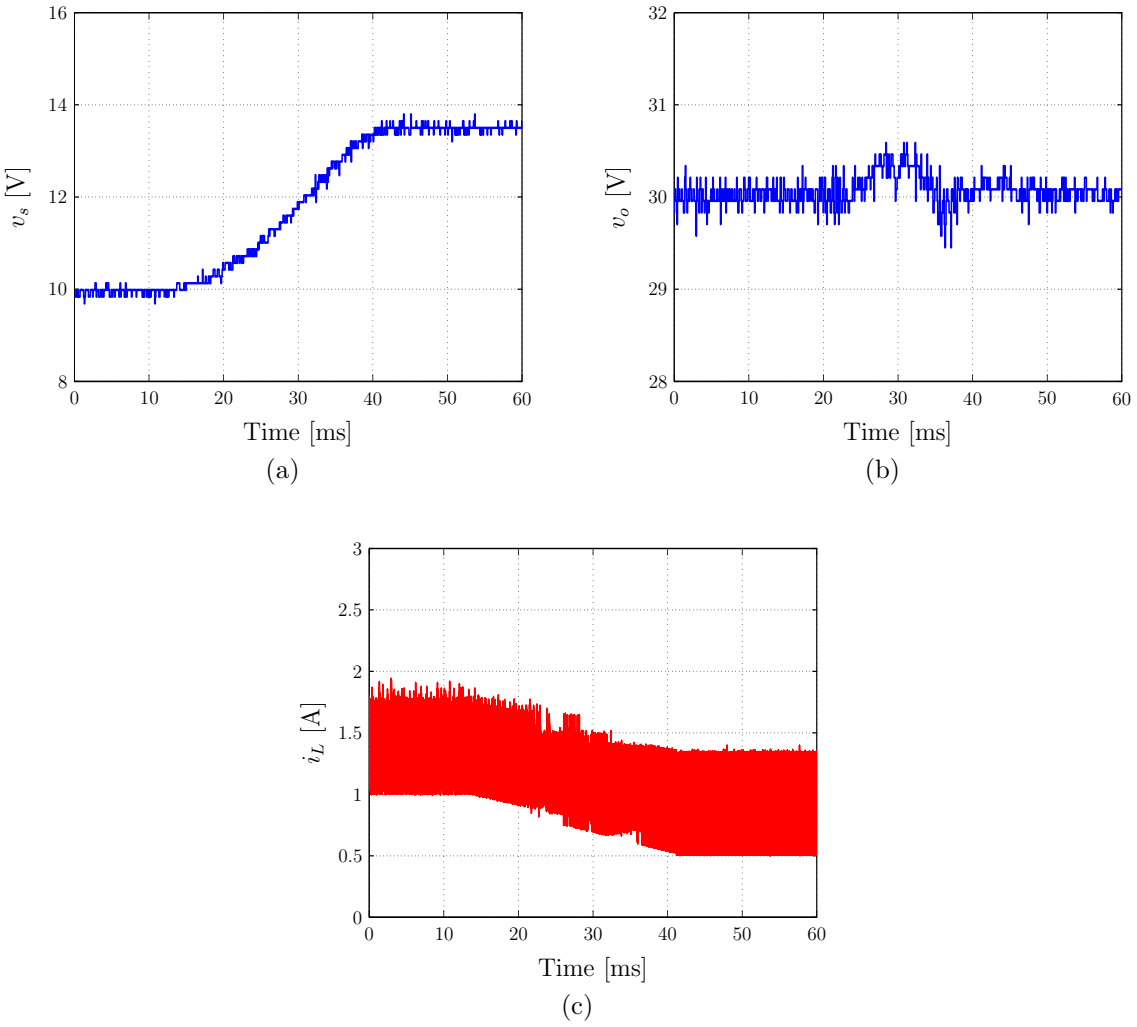


Figure 3.24: Experimental results for a ramp change in the input voltage: a) input voltage, b) output voltage, and c) inductor current.

occurring. The transient lasts for about $t \approx 3.5\,\text{ms}$.

3.5.1.3 Ramp Change in the Input Voltage

For the third case, a ramp change in the input voltage is imposed, starting at $t \approx 16\,\text{ms}$ and lasting until $t \approx 38\,\text{ms}$, as can be seen in Fig. 3.24(a). The input voltage is manually increased from $v_s = 10\,\text{V}$ to $v_s = 13.5\,\text{V}$, while the output voltage reference is $v_{o,\text{ref}} = 30\,\text{V}$. The effects on the output voltage and the inductor current are shown in Figs. 3.24(b) and 3.24(c), respectively. During this interval, the inductor current decreases until it reaches its new nominal value. The output voltage is not affected by the change in the input voltage and remains equal to its reference value.

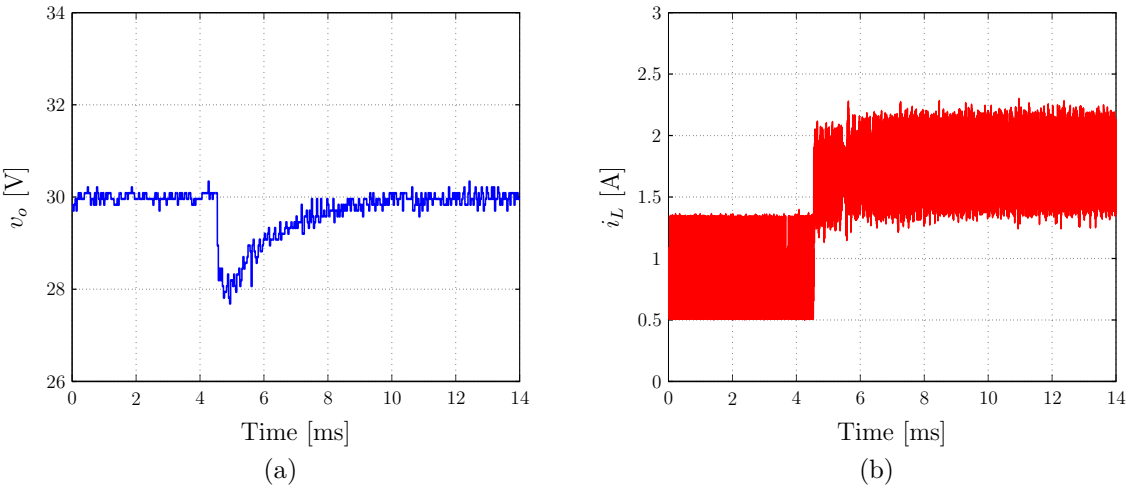


Figure 3.25: Experimental results for a step change in the load: a) output voltage, and b) inductor current.

3.5.1.4 Load Step Change

Finally, a step down in the load resistance is examined. At $t \approx 4.5\,\text{ms}$ the load resistance is halved, from its nominal value of $R = 73\,\Omega$ to $R = 36.5\,\Omega$. In Fig. 3.25 the closed-loop performance of the converter is depicted. The Kalman filter adjusts both the output voltage and the inductor current references. The average value of the current is instantaneously doubled, see Fig. 3.25(b), while a small undershoot in the output voltage is observed during the transient, see Fig. 3.25(a). When the converter reaches steady-state operation, a zero steady-state error is achieved thanks to the integrating character of the Kalman filter.

3.5.2 Voltage-Mode MPC

Due to computational restrictions imposed by the computational platform, a six-step prediction horizon was implemented, i.e. $N = 6$ and the sampling interval was set to $T_s = 10\,\mu\text{s}$. The prediction horizon was split into $N_1 = 4$ and $N_2 = 2$ with $n_s = 2$. The weight in the objective function was chosen as $\lambda = 0.5$. The covariance matrices of the Kalman filter are the same as in Section 3.4.2, i.e.

$$\boldsymbol{Q} = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 50 \end{bmatrix}, \quad \boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

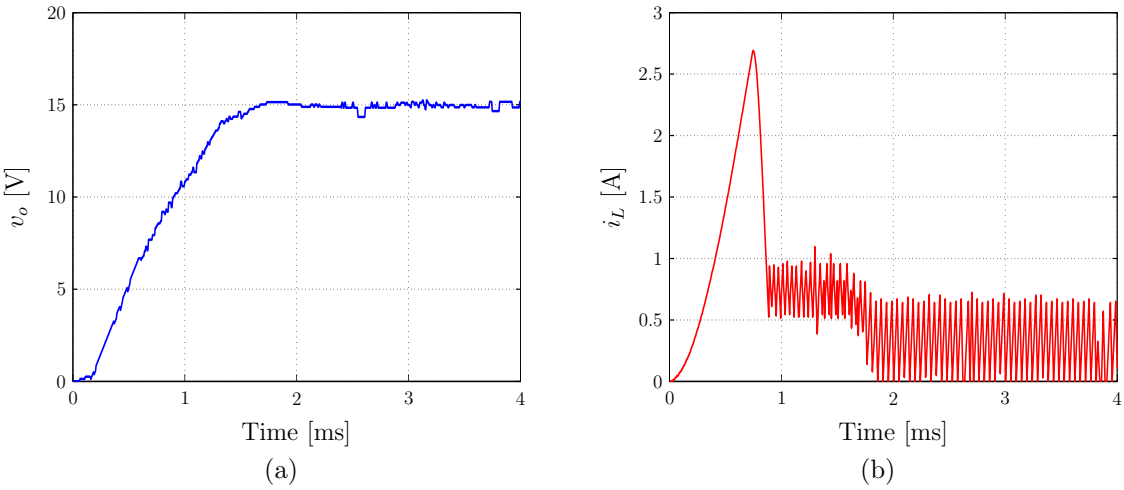


Figure 3.26: Experimental results for nominal start-up: a) output voltage, and b) inductor current.

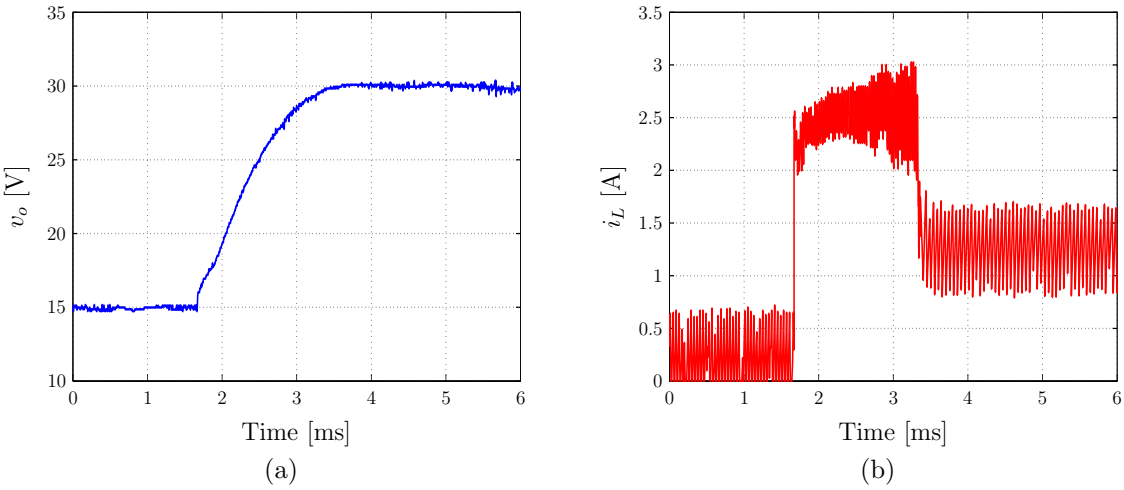


Figure 3.27: Experimental results for a step-up change in the output voltage reference: a) output voltage, and b) inductor current.

3.5.2.1 Nominal Start-Up

In Fig. 3.26 the output voltage and the inductor current of the converter are depicted during start-up. The inductor current rapidly increases to charge the output capacitor to the reference voltage level as fast as possible. The output voltage reaches its desired value in about $t \approx 1.8\,\text{ms}$. Subsequently, the inductor current reaches its nominal value and the converter operates in DCM.

3.5.2.2 Step Changes in the Output Reference Voltage

The second case to be analyzed is that of the transient behavior during step changes in the output reference voltage. A step-up change in the output reference voltage from

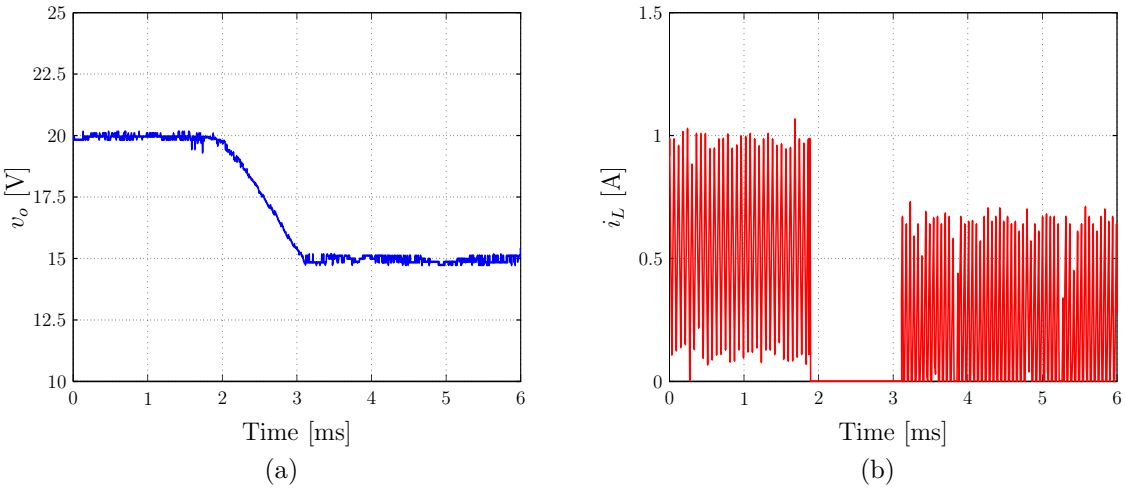


Figure 3.28: Experimental results for a step-down change in the output voltage reference: a) output voltage, and b) inductor current.

$v_{o,\text{ref}} = 15\,\text{V}$ to $v_{o,\text{ref}} = 30\,\text{V}$ occurs at $t \approx 1.7\,\text{ms}$. The response of the converter is illustrated in Fig. 3.27. The inductor current instantaneously increases, enabling the output voltage to reach its new desired level as fast as possible. This happens in about $t \approx 1.9\,\text{ms}$, without a significant overshoot.

Moreover, a step-down change, illustrated in Fig. 3.28, is investigated. The output reference voltage changes from $v_{o,\text{ref}} = 20\,\text{V}$ to $v_{o,\text{ref}} = 15\,\text{V}$ at $t \approx 1.9\,\text{ms}$. As can be seen, the controller exhibits a favorable performance; the inductor current is instantly reduced to zero so as to allow the capacitor to discharge through the resistor, and the converter reaches the new steady-state operating point in about $t \approx 1.2\,\text{ms}$.

3.5.2.3 Ramp Change in the Input Voltage

Subsequently, the input voltage is manually increased from $v_s = 10\,\text{V}$ to $v_s = 15\,\text{V}$ (the output reference voltage is $v_{o,\text{ref}} = 30\,\text{V}$), resulting in a voltage ramp from $t \approx 16\,\text{ms}$ until $t \approx 38\,\text{ms}$, see Fig. 3.29. During the transient, the inductor current changes accordingly in a ramp-like manner down to its new steady-state value. It can be seen that the output voltage remains unaffected and is kept equal to its reference value, implying that input voltage disturbances are very effectively rejected by the controller and the Kalman filter.

3.5.2.4 Load Step Change

The last case examined is that of a step-down change in the load resistance occurring at $t \approx 1.2\,\text{ms}$. With the converter operating at the previously attained operating point, the load resistance is halved, i.e. from $R = 73\,\Omega$ to $R = 36.5\,\Omega$. As can be observed in Fig. 3.30, the Kalman filter quickly adjusts the voltage reference accordingly, resulting

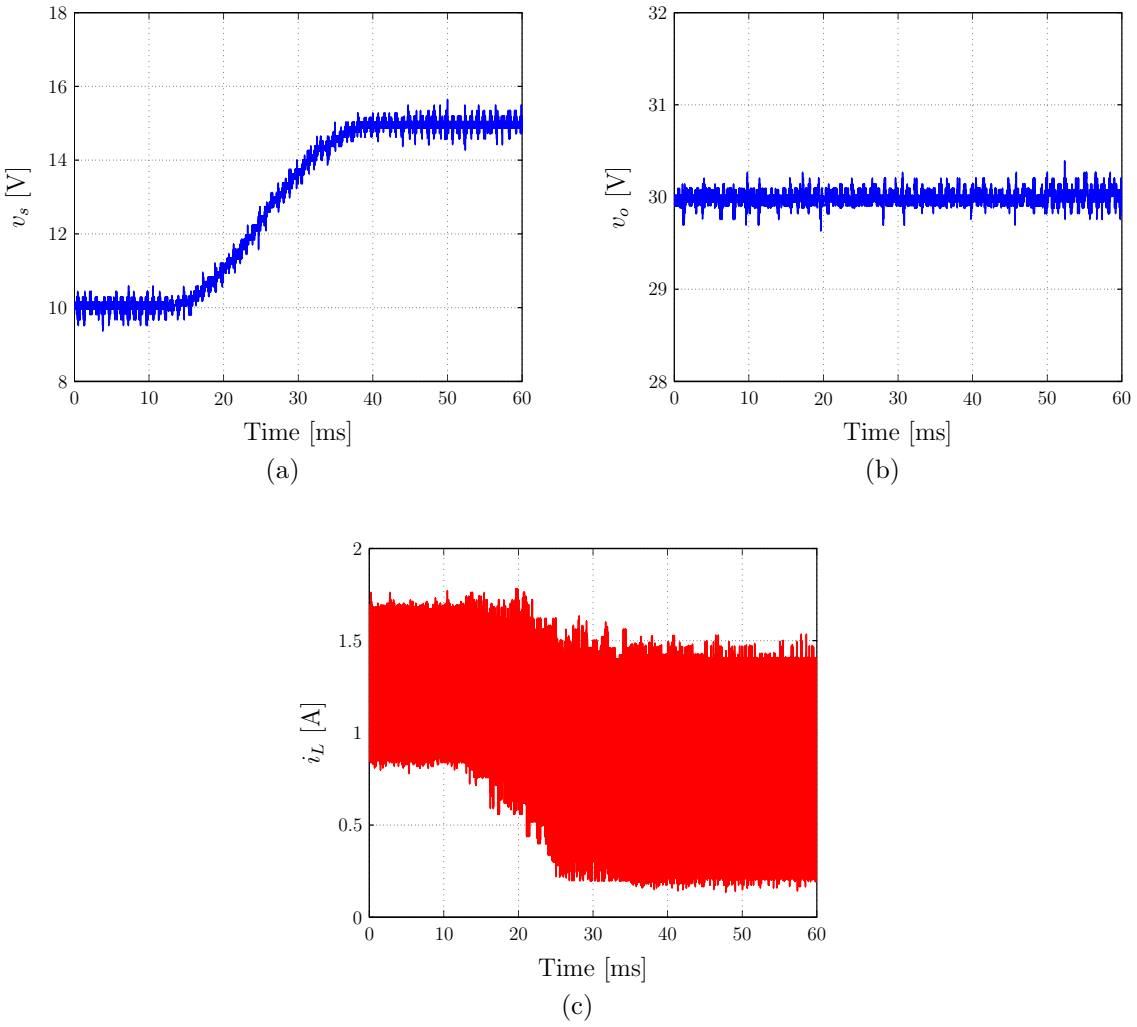


Figure 3.29: Experimental results for a ramp change in the input voltage: a) input voltage, b) output voltage, and c) inductor current.

in a zero steady-state error in the output voltage, thanks to its integrating nature.

3.6 Conclusions

In this chapter, a current-mode and a voltage-mode controller formulated in the framework of model predictive control (MPC) have been proposed. The discrete-time model of the converter, used by both the current and the voltage controller, is designed such that it accurately predicts the plant behavior both when operating in continuous (CCM) as well as in discontinuous conduction mode (DCM). As a result, the formulated controller is applicable to the whole operating regime, rather than just to a particular operating point.

For the current-mode controller two different MPC approaches based on enumeration

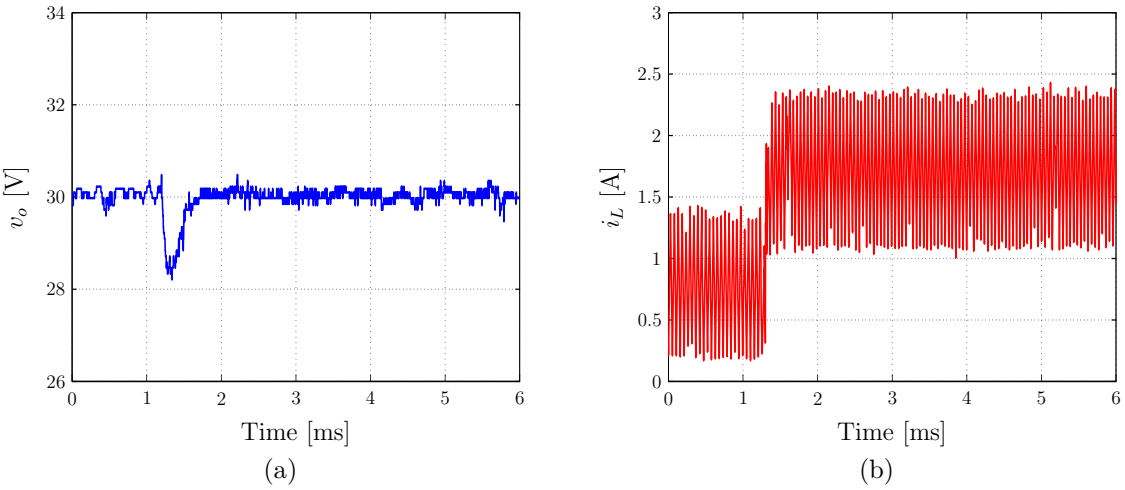


Figure 3.30: Experimental results for a step-down change in the load: a) output voltage, and b) inductor current.

have been introduced. The implementation of MPC as a current controller (rather than a voltage controller) enables the use of a relatively short prediction horizon, since the current exhibits a minimum-phase behavior with respect to the control input. Therefore, the required computational power is significantly reduced. The outer loop is augmented by a Kalman filter, suitable for all operating modes. This state estimation scheme is designed so as to cope with all possible disturbances and uncertainties, which might arise from real-world nonidealities. To this end, the controller aims at rejecting all disturbances, including load and input voltage variations. The performance of the proposed methods are compared via simulations. Both MPC approaches yield a similar favorable behavior during transients. Moreover, experimental results—for the average current error-based approach—are provided, validating the effectiveness of the proposed controller and a high degree of robustness to parameter variations.

For the voltage-mode controller, the proposed MPC approach based on enumeration aims to directly regulate the output voltage along its reference, without the use of an underlying current control loop. With MPC operating at the physical limits, the superior dynamical behavior of the controller is guaranteed. Since the converter model is included in the controller, the time-consuming tuning of controller gains is avoided. The computational complexity is somewhat pronounced, but kept at bay by using a move blocking scheme; with this strategy a significant reduction of the computations required is achieved and thus the real-time implementation of the controller is facilitated. A load estimation scheme, namely a discrete-time switched Kalman filter, is implemented to address time-varying and unknown loads and to ensure robustness to parameter variations; thanks to its integrating action, it provides offset-free tracking of the output voltage. Simulation and experimental results demonstrate the potential advantages of the proposed

methodology.

The proposed schemes carry several benefits. The very fast dynamics achieved by MPC, combined with its inherent robustness properties, are some of its key beneficial characteristics. Furthermore, thanks to the fact that the control objectives are expressed in the objective function in a straightforward manner, the design process is simple and laborious tuning is avoided. These benefits outweigh the drawbacks, which arise from the variable switching frequency due to the absence of a modulator and the direct manipulation of the converter switches.

Future Research

Working on the stability analysis of the derived optimal control schemes is certainly a challenging topic to be investigated. Furthermore, tuning the control effort term weighting factor in such a way so that a constant switching frequency results for a wide range of operating points is an open issue. Finally, another interesting point is to implement strategies to further reduce the computational complexity of the MPC-based algorithms.

Chapter 4

Cascaded H-Bridge Multilevel Rectifiers

4.1 Introduction

Multilevel converters are power converters that are widely used in the area of high-power medium-voltage energy control [121]. From oil, chemical and water plants to power generation, energy transmission and electric motor drive systems, multilevel converters have proved indispensable since they provide a cost-effective solution [39].

The basic concept of multilevel converters is to synthesize a sinusoidal voltage from several levels of voltages. To do so, a series of power semiconductor switches with several lower voltage dc sources—typically capacitor voltage sources—are used to synthesize a staircase voltage waveform. The more the voltage levels are, the closer to the sinusoidal voltage is the synthesized staircase voltage. Therefore, by appropriately turning on and off the power switches of the converter, different voltage levels result, since the multiple dc sources are aggregated in several ways. In this way, higher voltage levels are achieved.

From the above, it is straightforward to conclude that the widespread success of multilevel converters lies on the fact that they can achieve high power and voltage levels, while the rated voltage of the power switches is small, since it depends only upon the rating of the dc voltage sources to which they are connected. Hence, shorter turn-on and turn-off times are allowed compared to switches of higher voltage ratings, resulting in lower switching and conduction losses, and thus in an improved efficiency of the system.

However, this is not the only advantage of this class of converters, but they carry several additional attractive features compared to the conventional two-level converter. Since the multilevel staircase voltage approximates a sinusoidal waveform this means that the ac voltage is of good quality with low distortion. In addition, due to its staircase nature the dv/dt stresses are significantly reduced, and thus electromagnetic compatibility problems are avoided. Furthermore, multilevel converters draw ac currents with low

distortion, despite the fact that they can operate at very low switching frequency. Finally, assuming inverter operation, the stress in the bearings of a motor driven by a multilevel converter is reduced owing to the low common-mode voltage [25].

On the other hand there are some disadvantages. The most prominent drawback of multilevel converters is the increased number of power switches needed. Since each switch requires a related gate drive circuit the cost of the overall system increases. Furthermore, reliability issues may arise due to the greater number of switches. Finally, more complex control techniques are required, not only to overcome a potential fault in a switch element, but also to meet the more demanding operational requirements.

Nowadays, three different major multilevel converter structures exist: cascaded H-bridge (CHB) [52], neutral point clamped (NPC) [106], and flying capacitor (FC) [101]. Each of these topologies has its own characteristics, trade-offs, advantages and disadvantages. In the present chapter we focus on the CHB multilevel converter—operated as rectifier—and thus the analysis of the other two topologies is out of the scope of this chapter.

4.1.1 Control of CHB Multilevel Rectifiers

Among the multilevel converters, the CHB embodies the qualities of the most attractive topology in comparison to the NPC and the FC. The reasons for this are the reduced number of the switching devices, as well as its high modularity [119]. Furthermore, it has an increased fault tolerance due to the independent operation of the cascade-connected H-bridge modules (or cells).

However, several issues are still open, specifically, when the topology is operated as a multilevel rectifier. In this mode of operation, the CHB rectifier aims to achieve n—independently performing—isolated dc buses, resulting in the need for more complex control strategies. In addition, the converter has to operate always under unity power factor with minimum power losses, while at the same time respecting the operational limits imposed by the topology [134]. Thus, numerous research works have been reported in literature.

The standard approach to the control problem is to employ two loops. The outer—slow—loop, is the voltage loop, and the inner—fast—loop is the current loop. For the outer loop the conventional PI controllers are used to achieve the voltage regulation. One PI controller is used for each cell so as to regulate the respective output voltage to its desired level. The outputs of the PI controllers, in turn, determine the reference value of the input (ac) current, see Fig. 4.1.

For the inner loop several algorithms have been developed to meet most of the control goals. Contemporary research has engaged unique control schemes based on both linear and nonlinear techniques. From the current loop the control signal is derived, which is

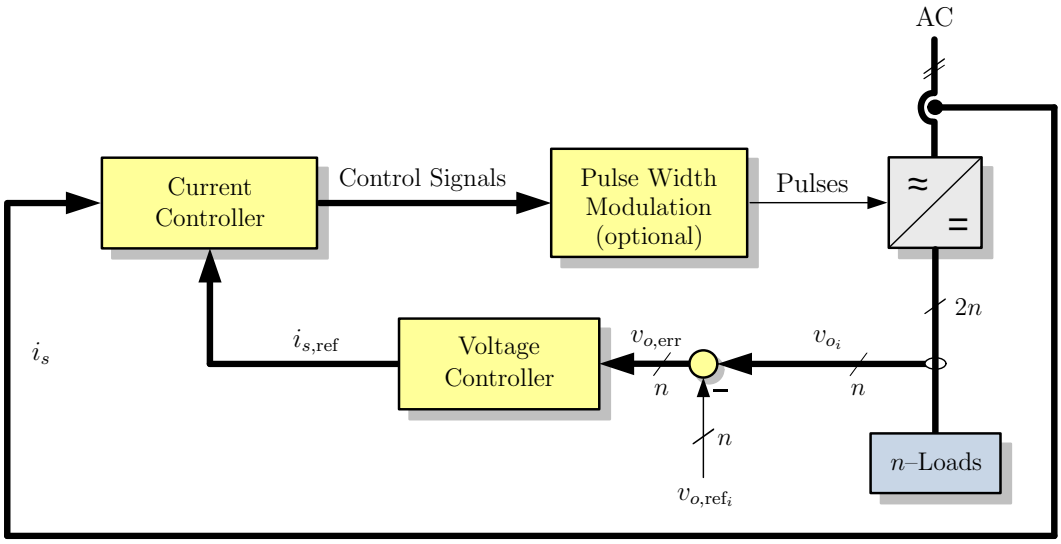


Figure 4.1: Current control block diagram of CHB multilevel rectifier ($i \in \{1, 2, \ldots, n\}$).

forwarded either to a modulator in order to control the pulses of the converter switches, or, in the absence of a modulator, directly to the switches (Fig. 4.1). However, in the majority of the introduced schemes a modulator is employed since it gives the benefit of constant switching frequency. A high percentage of the proposed modulation strategies rely on the multicarrier approach (multicarrier PWM—MPWM) [55,76,78,87], while others use conventional [86] or generalized [85] modulation methods with low computational complexity exhibiting noteworthy performance.

Linear, PI-based control schemes are developed for the CHB multilevel rectifier in [22, 33]. In [22] the presence of a PI controller results in a steady-state error since the controlled signal (the input current) is an ac signal. Furthermore, the proposed strategy is not able to operate under asymmetrical voltage potentials. To overcome this, in [33] a proportional-resonant (PR) controller is used instead; the steady-state error is eliminated, and the dynamic performance of the controller is improved. Furthermore, a phase-shift PWM (PSPWM) [129] technique is used to reduce the harmonics in the multilevel voltage[1]. In addition, an adequate performance under unbalanced output cell voltages is achieved. Nevertheless, the inherent disadvantages of linear controllers still exist: since controllers of this type are usually tuned to achieve the desirable performance only over a narrow operating range, the performance outside this range deteriorates significantly.

For improved robustness a hysteresis current controller is implemented in [91] and in [60]. The method provides fast dynamics and zero steady-state error. However, the direct manipulation of the switches results in a variable switching frequency. Besides that, the proposed technique does not achieve decoupling of the cells, and thus it is not

[1]PSPWM is a modulation technique which is based on the classic carrier-based sinusoidal PWM (SPWM) that uses phase shifted modulation signals.

suitable for operation under uneven output cell voltages and unbalanced loads.

A different nonlinear approach that yields similar satisfactory results in terms of robustness is designed in the framework of sliding mode control [62]. The decoupling of the H-bridge cells is achieved, and therefore the independent control of each module, i.e. the desired regulation of the dc voltages is attained even in cases of different voltage references and loads. For this strategy a modulator is not required, thereby the converter operates with a variable switching frequency. Furthermore, the control design approach generates aggressive control effort to reduce the tracking error, resulting in significant output voltage over- and undershoots during transients.

For the estimation of the model parameters an adaptive-passivity control is introduced in [23]. The proposed scheme is capable of regulating the voltage under asymmetric operating conditions due to the load estimation of each module. Moreover, model mismatches and uncertainties due to real-world nonidealities are successfully tackled. Hence, the robustness of the controller is guaranteed even if the mismatches between the theoretical and actual parameter values are substantial. Nonetheless, a high switching frequency is required, resulting in increased switching losses, while the computational burden is particularly high.

On the other hand, for switching frequency reduction and power losses minimization selective harmonic elimination PWM (SHE-PWM) control [135] is very promising candidate. This strategy ensures that the dc-link capacitor voltages for each cell of the converter are always balanced, even when the loads of the individual cells are not. This implies that the algorithm is not applicable to asymmetrical voltage potential.

Despite the effectiveness of the existing control approaches, there are still open tasks such as ease of controller design and elimination of tuning. Furthermore, the majority of the proposed strategies are suitable for a limited range of operation, i.e. balanced output cell voltages and loads. Therefore, in order to successfully tackle these issues—and thanks to the rapid development of fast microprocessors—computationally demanding MPC-based algorithms have been implemented [32,131,133,138], as well. Despite the fact that in [32, 131] the CHB converter is operated as an inverter the main control objectives are the same with those for a rectifier operation. The MPC-based strategies exhibit satisfactory performance, with fast transient responses, and reduced computational effort, since switching constraints are taken into account. However, a one-step prediction horizon is used, meaning that stability issues may arise. A one-step prediction horizon is employed in [133, 138], too, where the CHB converter is operated as a rectifier, thus the same problems are present. In addition, the formulated objective function in [133] consists of many different terms, i.e. laborious tuning is required in order to approximate the trade-offs between these terms.

In this work, an MPC strategy is adapted to the CHB multilevel rectifier consisting

of n cells. By directly manipulating the switches of the converter the regulation of the sinusoidal input current to its reference is achieved, and it is kept in phase with the supply voltage. Furthermore, the proposed control scheme aims to achieve independent voltage regulation of the H-bridge cells. A voltage term is added in the objective function that maintains and improves the effectiveness of the strategy introduced under transient operating conditions; the proposed algorithm exhibits favorable performance during transients. Moreover, the controller is able to stabilize the system for the entire operating regime due to the exhaustive search of all the possible switching combinations. These benefits overshadow the drawbacks of the proposed technique such as the increased computational complexity and the variable switching frequency resulting from the absence of a modulator. However, a method to impose constraints on the switching transitions is proposed; the number of the transitions to be evaluated in real-time is reduced, and thus the computational effort required.

4.2 Model of the Cascaded H-Bridge Multilevel Rectifier

4.2.1 Continuous-Time Model

The topology of the CHB rectifier with n cells connected in series is illustrated in Fig. 4.2. The ac side consists of a boost inductance L, with internal resistor R_L. At the dc side each cell consists of a filter capacitor C_{o_i}, where $i \in \{1, 2, \ldots, n\}$ denotes the number of the cell, connected in parallel with the load.

Each H-bridge cell is composed of four switches S_{ij}[2], where $j \in \{1, \ldots, 4\}$ refers to the respective switch of the cell. The switches of each cell operate dually and in pairs denoted by T_{ip}, with $p \in \{1, 2\}$; the odd indexed switches (S_{i1} & S_{i3}) form one pair ($p = 1$) and the even indexed (S_{i2} & S_{i4}) the other ($p = 2$). The possible switching combinations of the ith cell of the converter are: $T_{i1}T_{i2} = 10$, $T_{i1}T_{i2} = 00$, $T_{i1}T_{i2} = 01$ and $T_{i1}T_{i2} = 11$, where "0" denotes the *off* state of the upper switch of the pair and "1" the *on* state. In Table 4.1 the switching combinations for an n-cell CHB rectifier and the resulting reflected multilevel voltage to the ac side are summarized. The symbolism "††" stands for the case where the switching states T_{ip} of both pairs of the ith cell are the same, i.e. $T_{i1}T_{i2} = 00$ or $T_{i1}T_{i2} = 11$. Furthermore, in Table 4.2 the switching combinations and the corresponding level of the voltage v_{ab} for a two-cell CHB rectifier are summarized.

The state-space representation of the CHB multilevel rectifier in the continuous-time

[2]Usually each switch is composed of an IGBT and an anti-parallel freewheeling diode.

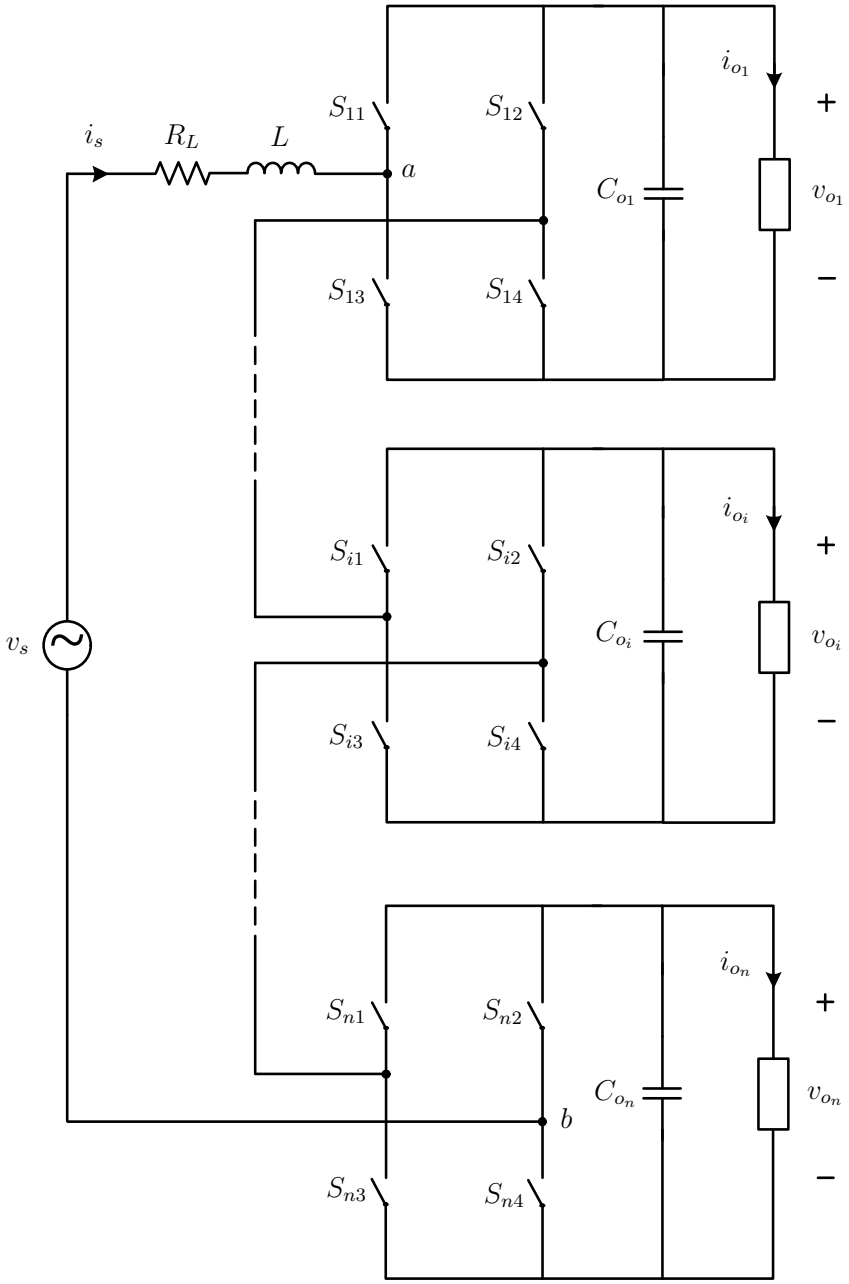


Figure 4.2: Topology of the single-phase CHB multilevel rectifier consisting of n cells connected in series.

domain is of the form

$$\frac{d\boldsymbol{x}(t)}{dt} = \big(\boldsymbol{A}_1 + \boldsymbol{A}_2\boldsymbol{u}(t)\big)\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{w}(k) \tag{4.1a}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)\,, \tag{4.1b}$$

where

$$\boldsymbol{x}(t) = \Big[i_s(t) \qquad v_{o_1}(t) \qquad \dots \qquad v_{o_n}(t)\Big]^T, \tag{4.2}$$

is the state vector, encompassing the inductor current and the output voltages of the

Table 4.1: Switching States of a Single-Phase Cascaded H-Bridge Multilevel Rectifier Consisting of n Cells and ac Side Voltage v_{ab}

$T_{11}T_{12}$		$T_{i1}T_{i2}$		$T_{n1}T_{n2}$	v_{ab}
††	$\cdots$	10	$\cdots$	10	$\sum_{\zeta\in\mathcal{H}} v_{o_\zeta}$
10	$\cdots$	††	$\cdots$	01	$\sum_{\zeta\in\mathcal{H}} v_{o_\zeta} - \sum_{\xi\in\mathcal{L}} v_{o_\xi}$
††	$\cdots$	††	$\cdots$	††	$0,\ i \in \mathcal{O}$
††	$\cdots$	01	$\cdots$	01	$\sum_{\xi\in\mathcal{L}} -v_{o_\xi}$

$$\mathcal{H} = \{\zeta \in \mathbb{N}^* \mid \zeta \leq n,\ T_{\zeta 1}T_{\zeta 2} = 10\}$$

$$\mathcal{L} = \{\xi \in \mathbb{N}^* \mid \xi \leq n,\ T_{\xi 1}T_{\xi 2} = 01\}$$

$$\mathcal{O} = \{1, 2, \ldots, n\}$$

Table 4.2: Switching States of a Single-Phase Cascaded H-Bridge Multilevel Rectifier Consisting of Two Cells and ac Side Voltage v_{ab}

$T_{11}T_{12}$	$T_{21}T_{22}$	v_{ab}
10	10	$v_{o_1} + v_{o_2}$
10	††	v_{o_1}
††	10	v_{o_2}
10	01	$v_{o_1} - v_{o_2}$
01	10	$v_{o_2} - v_{o_1}$
††	††	0
††	01	$-v_{o_2}$
01	††	$-v_{o_1}$
01	01	$-v_{o_1} - v_{o_2}$

individual cells. The input matrix $\boldsymbol{u}(t) \in \mathbb{R}^{m\times m}$, with $m = n + 1$, is given by

$$\boldsymbol{u}(t) = \begin{bmatrix} d_{11} & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ d_{i1} & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ d_{n1} & 0 & \cdots & 0 & \cdots & 0 \\ 0 & d_{m2} & \cdots & d_{mi+1} & \cdots & d_{mn+1} \end{bmatrix}, \tag{4.3}$$

where the entries of the matrix are

$$d_{i1} = d_{mi+1} = u_{i1} - u_{i2}\,. \tag{4.4}$$

The binary variable $u_{ip} \in \{0, 1\}$ is introduced in order to model the switching state of each dually operated pair of switches T_{ip}; $u_{ip} = 1$ refers to the case where $T_{ip} = 1$, and $u_{ip} = 0$ to the case being $T_{ip} = 0$. The input voltage $v_s(t)$ and the load current $i_{o_i}(t)$ of each cell form the vector of the disturbances

$$\boldsymbol{w}(t) = \begin{bmatrix} v_s(t) & i_{o_1}(t) & \dots & i_{o_n}(t) \end{bmatrix}^T, \tag{4.5}$$

while the respective output voltages are considered as the output, i.e.

$$\boldsymbol{y}(t) = \begin{bmatrix} v_{o_1}(t) & \dots & v_{o_n}(t) \end{bmatrix}^T. \tag{4.6}$$

Finally the matrices $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{B} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{C} \in \mathbb{R}^{n\times m}$ are given by

$$\boldsymbol{A}_1 = \begin{bmatrix} -\frac{R_L}{L} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}, \tag{4.7}$$

$$\boldsymbol{A}_2 = \begin{bmatrix} 0 & 0 & \cdots & 0 & -\frac{1}{L} \\ \frac{1}{C_{o_1}} & 0 & \cdots & 0 & 0 \\ 0 & \frac{1}{C_{o_2}} & \ddots & \vdots & \vdots \\ \vdots & & \ddots & 0 & 0 \\ 0 & \cdots & 0 & \frac{1}{C_{o_n}} & 0 \end{bmatrix}, \tag{4.8}$$

$$\boldsymbol{B} = \begin{bmatrix} \frac{1}{L} & 0 & \cdots & 0 \\ 0 & -\frac{1}{C_{o_1}} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & -\frac{1}{C_{o_n}} \end{bmatrix}, \tag{4.9}$$

$$\boldsymbol{C} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}. \tag{4.10}$$

4.2.2 Discrete-Time Model

The MPC controller is built around the discrete-time state-space model of the converter. The discrete-time model suitable to serve as an internal prediction model for the controller is derived by discretizing the continuous-time model (4.1) using the forward Euler approximation approach. This yields:

$$\boldsymbol{x}(k+1) = \boldsymbol{A}_d(\boldsymbol{u})\boldsymbol{x}(k) + \boldsymbol{B}_d\boldsymbol{w}(k) \tag{4.11a}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}_d\boldsymbol{x}(k)\,. \tag{4.11b}$$

The matrices are $\boldsymbol{A}_d(\boldsymbol{u}) = (\boldsymbol{I} + \boldsymbol{A}_1 T_s + \boldsymbol{A}_2 T_s \boldsymbol{u}(k))$, $\boldsymbol{B}_d = T_s\boldsymbol{B}$, and $\boldsymbol{C}_d = \boldsymbol{C}$, where $\boldsymbol{I}$ is the identity matrix and T_s is the sampling interval, and the matrices $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are detailed in Section 4.2.1.

4.3 Optimal Control of CHB Multilevel Rectifiers

In this section an MPC scheme for the CHB multilevel rectifier is introduced. The variables of concern are controlled by directly manipulating the switches of each cell, thus a modulator is not required.

For the CHB multilevel rectifier the control objectives are multiple and of equivalent importance. Firstly, the input current i_s of the topology should be sinusoidal and in phase with the supply voltage v_s, resulting in a unity power factor. Furthermore, the harmonic content of the current should be kept as low as possible, with a low total harmonic distortion (THD), while simultaneously the switching frequency should remain low in order to reduce the switching losses. Finally, the output voltage of each cell v_{o_i} should accurately track its reference, and remain unaffected by changes in the load.

4.3.1 Objective Function

The chosen objective function to be minimized in real-time is:

$$J(k) = \sum_{\ell=k}^{k+N-1} \Big(||i_{s,\mathrm{err}}(\ell+1|k)||_1 + ||v_{o,\mathrm{err}}(\ell+1|k)||_1 + \lambda||\Delta\boldsymbol{u}(\ell|k)||_1 \Big)\,, \tag{4.12}$$

which penalizes the evolution of the variables of concern over the finite prediction horizon N using the 1-norm (sum of absolute values).

The first term in (4.12) stands for the input current error. In the control method introduced here, the inner loop aims to regulate the inductor current to its reference, derived from the outer loop. Therefore, the respective deviation is taken into account, given by

$$i_{s,\mathrm{err}}(k) = i_{s,\mathrm{ref}} - i_s(k)\,. \tag{4.13}$$

The second term defined as

$$v_{o,\mathrm{err}}(k) = \frac{1}{n}\sum_{i=1}^{n} \left| v_{o,\mathrm{ref}_i} - \bar{v}_{o_i}(k) \right| , \tag{4.14}$$

is added to ensure the regulation of the output voltages of the rectifier cells to their references even when they are of different levels. In (4.14) $\bar{v}_{o_i}$ is the dc component of the output voltage of the ith cell, i.e.

$$\bar{v}_{o_i}(k) = \frac{1}{M}\sum_{l=0}^{M-1} v_{o_i}(k-l) , \tag{4.15}$$

where $M \in \mathbb{N}^*$ is the number of samples in one period of the second harmonic (relative to the input voltage frequency) of the output voltage, i.e. $2MT_s = T$, with T being the period of the input voltage. This means that when the rectifier operates under steady-state conditions—assuming accurate regulation of the output voltage of each cell to its reference—the voltage error given by (4.14) tends to zero. Thus, in steady-state operation the inner control loop is a current loop; current regulation suffices. Thereby, the objective function for steady-state operation is—approximately—of the form

$$J_{\text{st-st}}(k) = \sum_{\ell=k}^{k+N-1} \Big(||i_{s,\mathrm{err}}(\ell+1|k)||_1 + \lambda ||\Delta \boldsymbol{u}(\ell|k)||_1 \Big) . \tag{4.16}$$

On the other hand, under transient conditions the voltage term is "active"; it contributes to the improvement of the dynamic behavior of the system, since the controller aims to eliminate the nonzero voltage error by fast charging the capacitors C_{o_i}. Furthermore, augmented by the outer loop presented in Section 4.3.3, it ensures a zero steady-state voltage tracking error: when a difference between the actual and the reference voltage of one cell exists, the total cost, as it is calculated by (4.12), increases, thereby the controller should achieve both voltage and current tracking. However, in (4.14) the mean value of all the voltage errors is considered; the term $\frac{1}{n}$ is added so that the voltage error term will not significantly overshadow the current error term, and thus deactivating it. If the controller focuses only on the voltage error, then the current regulation will not be achieved, and then stability issues may arise[3].

Finally, the third term aims to decrease the switching frequency and to avoid excessive switching, by penalizing the difference between two consecutive switching states, i.e.

$$\Delta \boldsymbol{u}(k) = \boldsymbol{u}(k) - \boldsymbol{u}(k-1) . \tag{4.17}$$

The weighting factor $\lambda \in \mathbb{R}^+$ sets the trade-off between the current and the output voltage errors and the switching frequency f_{sw}. Some guidelines for tuning the weighting factor λ are presented in [31]. In Section 4.4 additional information on the impact of the weighting factor on the defined objective function are presented.

[3]This is due to the fact that the output voltage exhibits a nonminimum phase behavior with respect to the switching action.

4.3.2 Optimization Problem

The control input at time-instant kT_s is obtained by minimizing the objective function (4.12) over the optimization variable, which is the sequence of switching states over the horizon $\boldsymbol{U}(k) = [\boldsymbol{u}(k)\ \boldsymbol{u}(k+1)\dots\boldsymbol{u}(k+N-1)]^T$. Thus the following constrained optimization problem is formulated:

$$\begin{aligned} &\text{minimize} && J(k) \\ &\text{subject to} && (4.11)\,. \end{aligned} \tag{4.18}$$

The underlying optimization problem is a mixed-integer optimization problem [12]. For solving such type of problems enumeration is a straightforward option (see Section 2.4). By taking into account all possible combinations of the switching states ($u_{ip} = 0$ or $u_{ip} = 1$) the switching sequences to be examined are created. The evolution of the state is calculated based on (4.11a) for each of the 2^{2nN} sequences and the objective function is evaluated. The sequence $\boldsymbol{U}^*$ with the smallest associated cost is considered as the optimal solution, given by

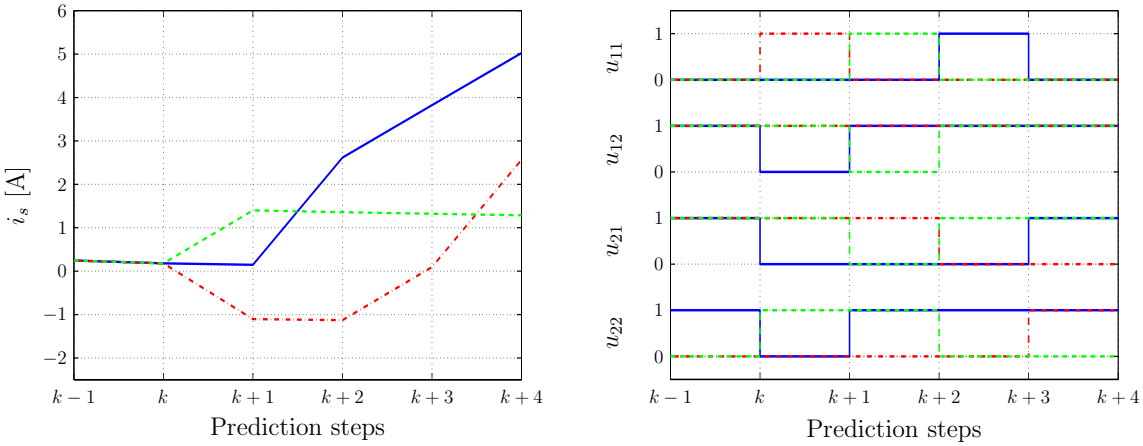
$$\boldsymbol{U}^*(k) = \arg\min J(k)\,. \tag{4.19}$$

Out of this sequence, the first element $\boldsymbol{u}^*(k)$ is applied to the converter; the procedure is repeated at $k+1$, based on new measurements acquired at the following sampling instance. An illustrative example of the predicted state—here the inductor current—and the sequence of the control actions, i.e. the switching state, is depicted in Fig. 4.3. Three candidate switching sequences are shown for the prediction horizon $N = 4$, and for a CHB rectifier consisting of two cells. In Fig. 4.3(a) the current of step k is the measured one, while from $k+1$ to $k+N$ the current evolution is depicted according to the switching sequences shown in Fig. 4.3(b).

4.3.3 Outer Loop

The outer loop is used for the voltage regulation. A PI controller is employed—one for each cell—to regulate the respective output voltage to its reference value. The input of the ith PI controller is the voltage error $v_{o,\mathrm{PI},\mathrm{err}_i} = v_{o,\mathrm{ref}_i} - v_{o_i}$ (see Fig. 4.4). The reference current $\hat{i}_{s,\mathrm{ref}}$ derived, shown in Fig. 4.4, is further synchronized with the supply voltage by a phase-locked loop (PLL), resulting in a sinusoidal reference current $i_{s,\mathrm{ref}}$.

The outer loop is tuned in such a way that the desired settling time and overshoot are achieved during start-up or step changes in the output reference voltage of a cell. In order to achieve a fast voltage regulation with as little overshoot as possible, the dynamics of the output voltages were registered under reference voltage step changes. The information acquired was used to adjust a simple first order model, and to select



(a) Predicted current trajectories (b) Predicted switching sequences

Figure 4.3: Three candidate switching sequences for a four-step prediction horizon, i.e. $N = 4$.

the gain parameters, k_{pr_i} and k_{int_i}, of the n-PI controllers[4]. With this procedure, the superior performance of the MPC-based inner loop is not deteriorated, as can be seen in Section 4.5.

4.3.4 Control Algorithm

The proposed control strategy is summarized in Algorithm 4.1, where the function f stands for the state-update given by (4.11).

The proposed control algorithm is shown in the block diagram in Fig. 4.4.

4.4 Discussion and Computational Complexity

4.4.1 Impact of Weighting Factor

As can be seen, in (4.12) the only term that needs to be tuned is the weighting factor λ. Therefore, the tuning procedure is significantly simplified. As mentioned in Section 4.3.1, λ is adjusted such that the desired switching frequency is achieved, by simultaneously taking the trade-off between tracking error and switching effort into account. However, the range of suitable values of λ varies. It depends mainly on: (1) the number of the cells of the converter, (2) the number of the steps in the prediction horizon, and (3) the operating point.

1. *Number of the cells of the converter*: The size of the input matrix $\boldsymbol{u}(k)$, given by (4.3), depends on the number of the cells. This means that for a converter

[4]The same values are used for the proportional gains k_{pr_i} of the n-PI controllers. The integral gains k_{int_i} are set at equal values, as well.

Algorithm 4.1 MPC algorithm

function $\boldsymbol{u}^*(k) = \text{CHBMPC}\ (\boldsymbol{x}(k), \boldsymbol{u}(k-1), \bar{v}_{o_i})$
  $J^*(k) = \infty$; $\boldsymbol{u}^*(k) = \emptyset$
  for all $\boldsymbol{U}$ over N **do**
    $J = 0$
    for $\ell = k$ to $k + N - 1$ **do**
      $\boldsymbol{x}(\ell + 1) = f(\boldsymbol{x}(\ell), \boldsymbol{u}(\ell))$
      $i_{s,\text{err}}(\ell) = i_{s,\text{ref}} - i_s(\ell + 1)$
      $v_{o,\text{err}}(\ell) = \frac{1}{n} \sum_{i=1}^{n} |v_{o,\text{ref}_i} - \bar{v}_{o_i}(\ell + 1)|$
      $\Delta\boldsymbol{u}(\ell) = \boldsymbol{u}(\ell) - \boldsymbol{u}(\ell - 1)$
      $J = J + |i_{L,\text{err}}(\ell)| + |v_{o,\text{err}}(\ell)| + \lambda|\Delta\boldsymbol{u}(\ell)|$
    end for
    if $J < J^*(k)$ **then**
      $J^*(k) = J$, $\boldsymbol{u}^*(k) = \boldsymbol{U}(1)$
    end if
  end for
end function

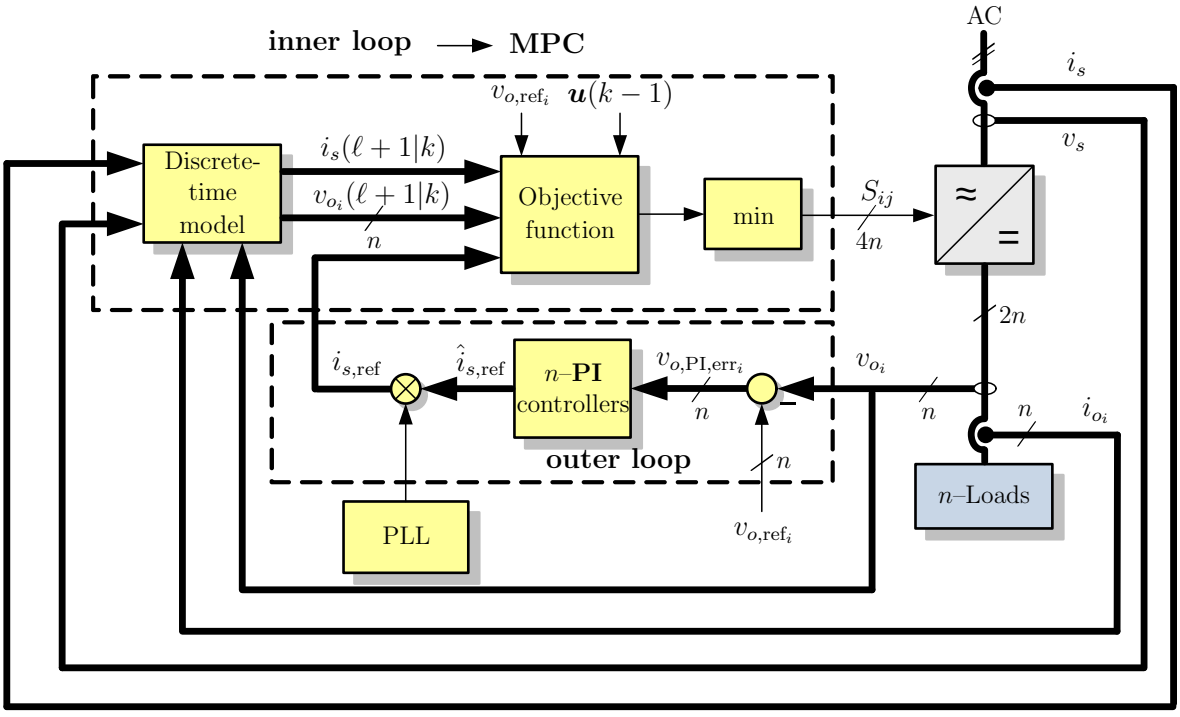


Figure 4.4: Block diagram of the proposed model predictive control (MPC) scheme.

consisting of many cells the contribution of the control effort term $\Delta\boldsymbol{u}$ to the objective function (4.12) is significant[5]. On the other hand, for a converter with a

[5]Note that the 1-norm—used in (4.12)—of a square matrix ($\Delta\boldsymbol{u}$ in (4.12)) is the maximum of the

smaller number of cells, the contribution of the control effort term to the objective function (4.12) is less.

2. *Number of the steps in the prediction horizon*: In (4.12) the switching transitions over the prediction horizon N are penalized. The more the prediction steps, the more the transitions that can take place within the horizon. Thus, the contribution of the control effort term to the objective function (4.12) varies based on the length of the prediction horizon. However, it should be mentioned that the contribution of the current term also varies[6].

3. *Operating point*: Usually, the weighting factor of the control effort term is tuned around the nominal operating point (see, for example, [31]). However, when the operating point changes, the contribution of the current error term to the objective function changes, too (keep in mind that the voltage term is zero under steady-state). On the other hand, the contribution of the control effort term $\Delta\boldsymbol{u}$ to the objective function (4.12) remains the same. This is due to the fact that the entries of the input matrix $\boldsymbol{u}(k)$ (4.3) do not depend on the operating point. According to (4.4), the nonzero entries of the matrix are the difference between the switching states of the upper switches of each cell of the converter. Therefore, these entries are in $\{-1, 0, +1\}$ regardless of the operating point. This means, that for different operating points and for the same value of the weighting factor, different switching frequencies may result.

To further investigate the impact of the weighting factor λ on the switching frequency and the output voltage error a case of a two-cell CHB rectifier operating under steady-state conditions is considered. The system parameters are shown in Table 4.3, while the prediction horizon is $N = 4$. As can be seen, the sampling interval is $T_s = 100\,\mu$s; this means that the maximum possible switching frequency is equal to $f_{sw,\max} = 5\,$kHz, i.e. $f_{sw,\max} = 1/(2T_s)$. However, in reality the switching frequency is much lower; the switches are not turned on and off every $2T_s$.

In Fig. 4.5 the output voltage error given by (4.14) and the switching frequency f_{sw} are depicted. As can be observed, an increase in the weighting factor causes a reduction in the switching frequency. However, for values of λ greater than $\lambda \approx 1.1$ a steady-state voltage error occurs. This is due to the fact that the controller puts more effort into penalizing

absolute column sums. Thus, the larger the number of the cells, the more the entries in each column of the input matrix. Thereby, the sum of the entries in the first column, i.e. the column with the largest number of nonzero entries, increases.

[6]The assumption that the voltage term is zero under steady-state operation is made, as explained in the text (see Section 4.3.1), i.e. the objective function (4.12) is replaced—without loss of generality—by (4.16).

Table 4.3: System Parameters

Parameter		Value
Number of cells	n	2
Rated power	P	1 kW
Nominal frequency	f	50 Hz
Input voltage	v_s	110 V rms
Boost inductance	L	8 mH
Internal resistance	R_L	0.7 Ω
Filter capacitance	C_{o_i}	2.2 mF
Sampling interval	T_s	100 μs

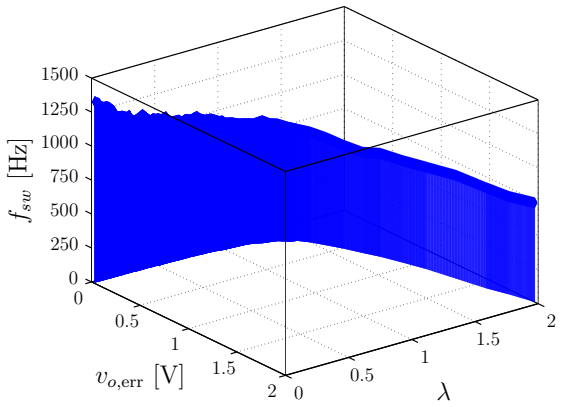


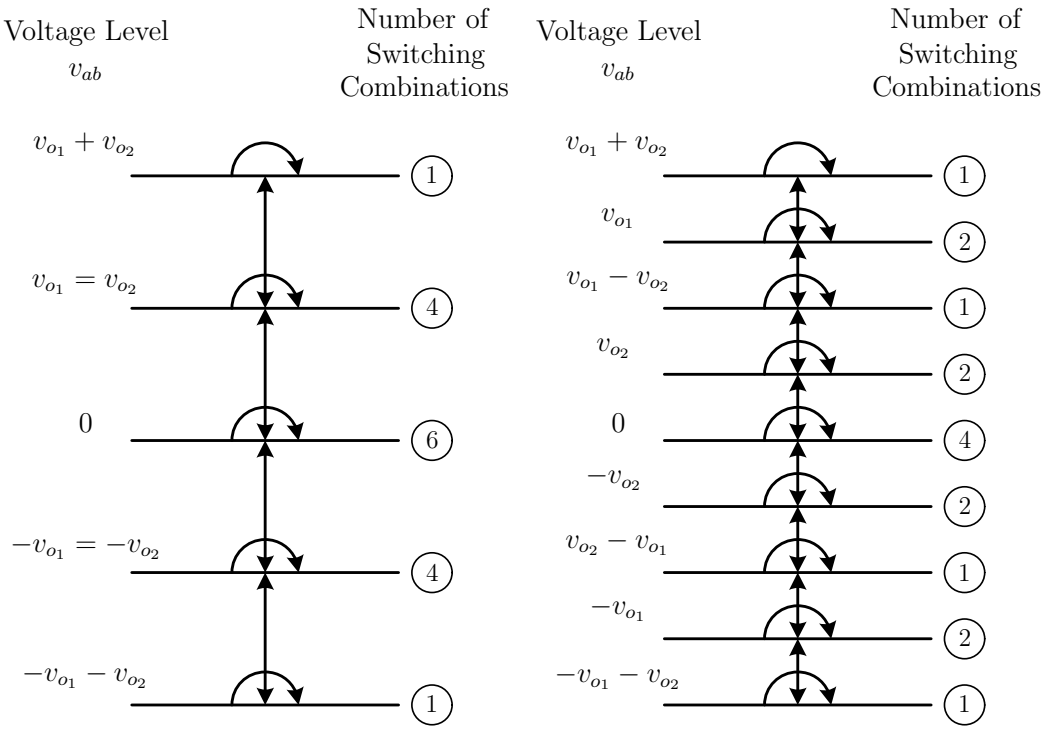
Figure 4.5: The output voltage error $v_{o,\text{err}}$ and the corresponding switching frequency f_{sw} versus the weighting factor λ when the converter operates under nominal conditions.

the switching transitions, rather than minimizing the input current and output voltages errors.

4.4.2 Switching Constraints

As already mentioned, the controller introduced here takes into account 2^{2nN} sequences, generated by all the possible switching combinations, in order to select the optimal one $\boldsymbol{U}^*$. In reality, however, when the converter operates under steady-state conditions not all the transitions from one switching state to another are possible. Hence, constraints could be posed to the switching transitions in order to trim the number of the examined switching sequences, resulting in a reduced computational effort.

The constraints are imposed by considering the multilevel waveform of the voltage v_{ab}

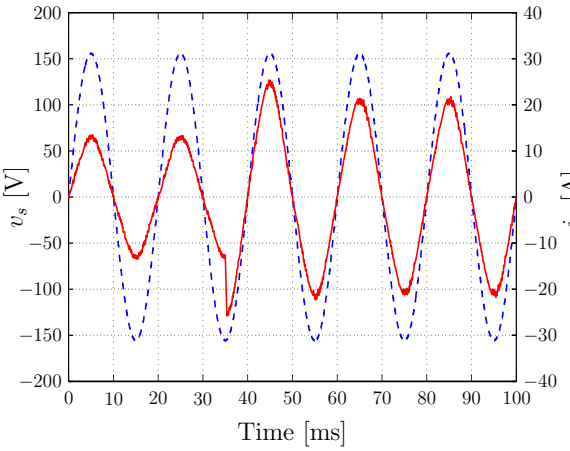


(a) Five-level v_{ab}. (b) Nine-level v_{ab}.

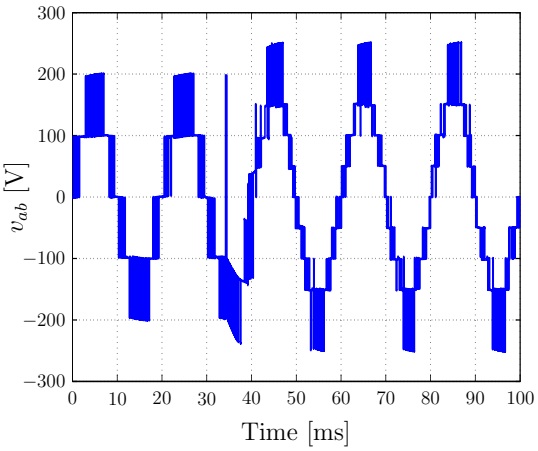
Figure 4.6: Allowable switching transitions in a two-cell CHB rectifier when the cells operate (a) at the same voltage potential and (b) at different potentials.

in the ac side converter terminals (see Fig. 4.2). As can be seen in Table 4.1, the total voltage levels of v_{ab} are $2n+1$ when the cell voltages are equal. These levels depend on the switching state of the cells, i.e. the way that the output voltage of each cell is reflected to the ac side. Hence, only these switching sequences that ensure smooth transition from one level to the neighboring one (lower or higher) are considered *feasible* and examined. Furthermore, when the converter is operating under unbalanced output cell voltages, the number of the voltage levels of v_{ab} increases, depending on the number of the different potentials. In such case the redundant states are significantly decreased, resulting in a further reduction of the computational complexity; the switching sequences that guarantee smooth transition from one voltage level to the next one are fewer.

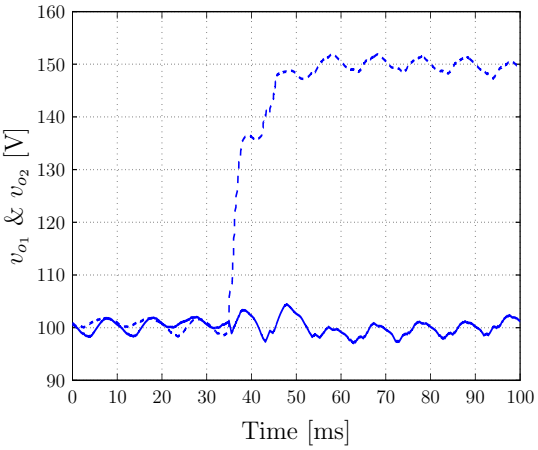
Fig. 4.6 shows an example of the allowable switching transitions for the case of a two-cell CHB rectifier. In Fig. 4.6(a) the converter operates under balanced output voltages, i.e. a five-level voltage waveform v_{ab} is produced. As can be seen in Table 4.2, 6 different switching states can produce the zero-voltage level. Thus, the maximum number of switching sequences to be examined corresponds to the case of $v_{ab}(k) = 0$. Assuming a one-step horizon the number of the possible optimal sequences is equal to 14: 6 sequences lead to a transition $v_{ab}(k) = 0 \rightarrow v_{ab}(k+1) = 0$, 4 sequences to a transition $v_{ab}(k) = 0 \rightarrow v_{ab}(k+1) = v_{o_1} = v_{o_2}$, and 4 sequences to a transition



(a) Input voltage (dashed line) and current (solid line).



(b) Ac side voltage.

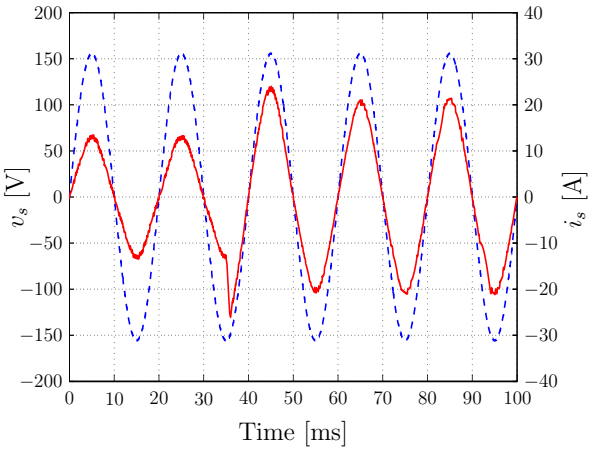


(c) Output voltage of first (solid line) and second cell (dashed line).

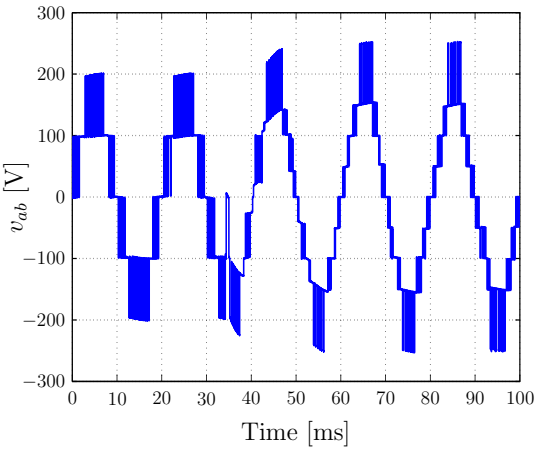
Figure 4.7: Without considering the switching constraints: Transient response of a two-cell CHB rectifier to a step-up change in the output voltage reference of the second cell (simulation results).

$v_{ab}(k) = 0 \rightarrow v_{ab}(k+1) = -v_{o_1} = -v_{o_2}$ (see Table 4.2). For the case of a two-step horizon, again the most computational effort is required when $v_{ab}(k+1) = 0$; 14^2 sequences should be examined. By extending the prediction horizon to N-steps, the worst-case scenario is when $v_{ab}(k) = 0 = v_{ab}(k+1) = \ldots = v_{ab}(k+N)$, corresponding to 14^N sequences. On the other hand, when the switching constraints are not considered the feasible sequences are $2^{2 \cdot 2N} = 16^N$. Following the same procedure, it can be shown that when the switching constraints are active the number of the sequences examined for the case of an n-level CHB rectifier is reduced, compared to the respective number of the unconstrained case.

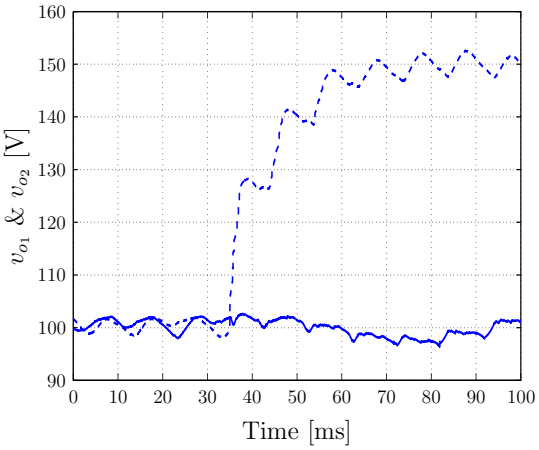
In Fig. 4.6(b) the allowable transitions in a nine-level waveform v_{ab}, resulting from the unbalanced cell voltages of a two-cell CHB converter, are depicted (it is assumed



(a) Input voltage (dashed line) and current (solid line).



(b) Ac side voltage.



(c) Output voltage of first (solid line) and second cell (dashed line).

Figure 4.8: Considering the switching constraints: Transient response of a two-cell CHB rectifier to a step-up change in the output voltage reference of the second cell (simulation results).

that $v_{o_2} < \frac{v_{o_1}}{2}$). Under these operating conditions the number of the feasible sequences is further reduced, since the redundancies are not that many. Once again, more redundant states correspond to the zero-voltage level compared to the other voltage levels. Therefore, following the same approach as before, in a one-step horizon the maximum feasible sequences are 8: 4 sequences for the transition $v_{ab}(k) = 0 \rightarrow v_{ab}(k+1) = 0$, 2 sequences for $v_{ab}(k) = 0 \rightarrow v_{ab}(k+1) = v_{o_2}$, and 2 sequences for $v_{ab}(k) = 0 \rightarrow v_{ab}(k+1) = -v_{o_2}$ (Table 4.2). For a N-step horizon the maximum switching sequences to be examined are 8^N, far fewer than the 16^N sequences encountered when the switching constraints are not considered.

However, the reduced computational complexity comes at a cost: the transient re-

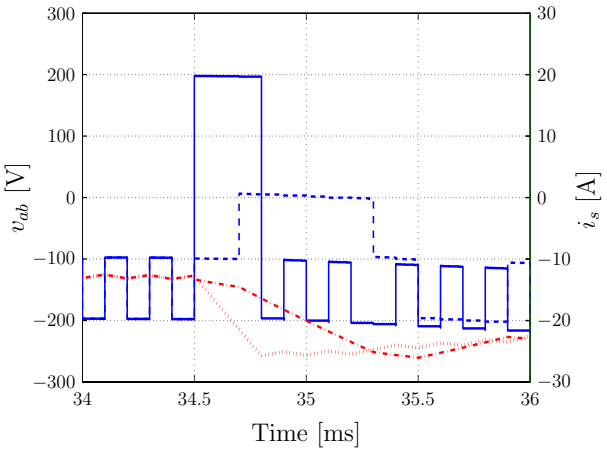
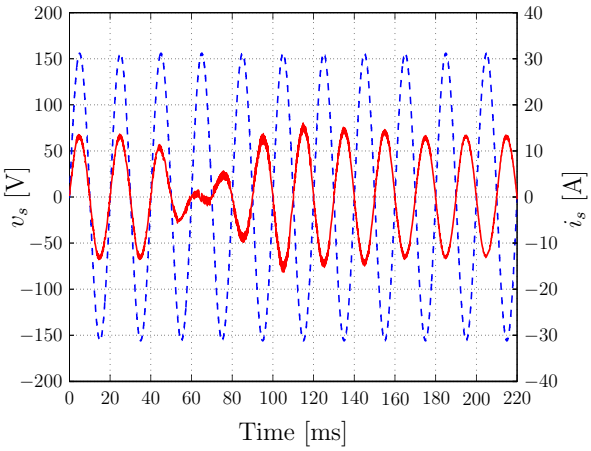


Figure 4.9: Detail of the ac side voltage without the switching constraints (solid line) and with them (dashed line), and of the input current without the switching constraints (dotted line) and with them (dash-dotted line) when the step change in the output voltage reference occurs.

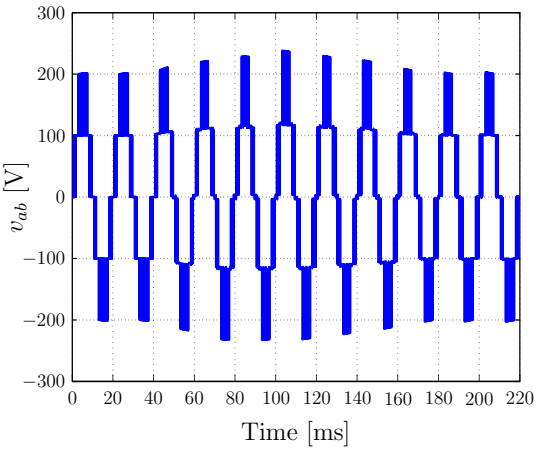
sponse of the system is deteriorated. This can be seen in Figs. 4.7 and 4.8, where a step-up change in the output voltage reference of the second cell of a two-cell CHB rectifier occurs at $t \approx 35\,\text{ms}$, from $v_{o,\text{ref}_2} = 100\,\text{V}$ to $v_{o,\text{ref}_2} = 150\,\text{V}$; the reference voltage of the first cell is $v_{o,\text{ref}_1} = 100\,\text{V}$ (the parameters of the system are shown in Table 4.3).

When the switching constraints are not taken into account (Fig. 4.7), the voltage of the second cell reaches its reference in about $t \approx 20\,\text{ms}$ (Fig. 4.7(c)). Due to the fact that there are no restrictions on the switching transitions, these switching states are applied that allow the instantaneous change in v_{ab} from its lowest voltage level $\{-v_{o_1} - v_{o_2}\}$, to its highest $\{v_{o_1} + v_{o_2}\}$ (Fig. 4.7(b)). This change results in a high di_s/dt, and consequently in a fast capacitor charging, see Fig. 4.7(a).

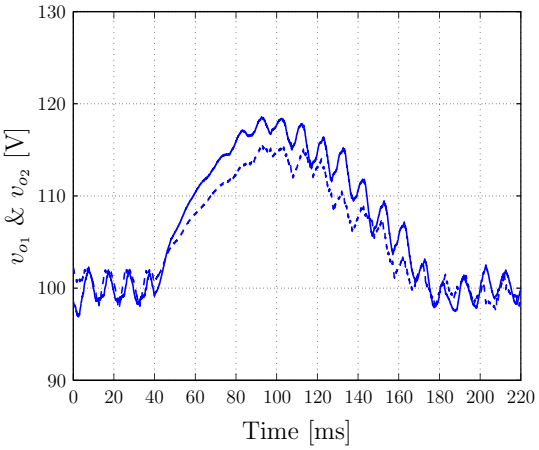
On the other hand, when the switching constraints are active (Fig. 4.8), the transient lasts more; the voltage of the second cell reaches its reference in about $t \approx 40\,\text{ms}$ (Fig. 4.8(c)), while a small undershoot in the voltage of the first cell is observed. For this case the current increases slower (Fig. 4.8(a)); the current slope di_s/dt is lower due to the fact that the switching states that lead to an immediate transition from level $\{-v_{o_1} - v_{o_2}\}$ of the multilevel voltage v_{ab} to level $\{v_{o_1} + v_{o_2}\}$ are not allowed. The ac side reflected voltage is initially equal to $\{-v_{o_1} - v_{o_2}\}$. Following, the only permitted switching transition generates a voltage v_{ab} equal to $\{-v_{o_1} = -v_{o_2}\}$. Finally, since the goal is to increase the input current, a switching state is selected that results in a further decrease in the multilevel voltage to the next allowable level, i.e. the zero-voltage level. In Fig. 4.9 the multilevel voltage v_{ab} and the input current i_s from both implementations—with and without the switching constraints—are shown in detail at the beginning of the transient. Finally, it should be noted that if the high input current during the transients is a concern, a current limit can be added, with the trade-off of higher settling time.



(a) Input voltage (dashed line) and current (solid line).



(b) Ac side voltage.



(c) Output voltage of first (solid line) and second cell (dashed line).

Figure 4.10: Simulation results of a two-cell CHB rectifier operating under normal (for $t < 40\,\text{ms}$) and regenerative conditions (for $t > 40\,\text{ms}$).

4.4.3 Regeneration Mode

An additional feature of the proposed MPC strategy is its ability to fulfill the control objectives even when the converter is operating in regenerative mode, i.e. when the load delivers power to the supply. In order to investigate the performance of the proposed strategy under regenerative load conditions a two-cell CHB rectifier, the parameters of which are shown in Table 4.3, is considered. The system is operating under nominal conditions and balanced loads; the output voltage reference values are $v_{o,\text{ref}_1} = v_{o,\text{ref}_2} = 100\,\text{V}$, i.e. the load current of each cell is $i_{o_i} = 5\,\text{A}$. In order to model the load current a 5-A current source is connected to each cell. Finally, a two-step prediction horizon is used, while the switching constraints are not taken into account.

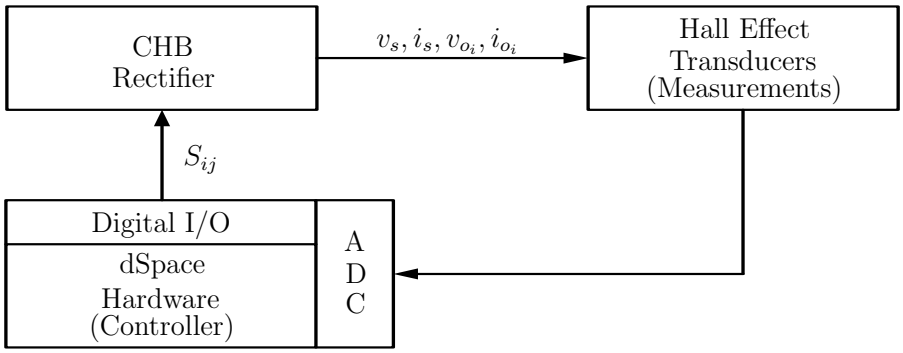


Figure 4.11: Schematic of the experimental setup. (ADC stands for the analog-to-digital converter.)

In Fig. 4.10 the transition from motoring to generating mode is shown. At time $t = 40\,\text{ms}$ the direction of the current flow is reversed to both cells so as to change the power flow from the cells to the grid. This forces the inductor current to change its polarity; the input current is 180° out of phase with respect to the supply voltage, as can be seen in Fig. 4.10(a). Furthermore, after an initial increase because of the power delivered by the loads, the output voltages of the cells, v_{o_1} and v_{o_2}, accurately track their reference values, see Fig. 4.10(c).

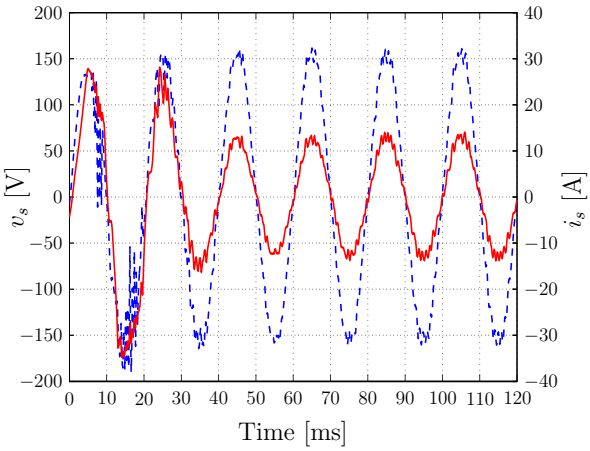
4.5 Experimental Results

In this section experimental results of the proposed control algorithm are presented. As a case study a CHB single-phase rectifier consisting of two H-bridge cells is considered, i.e. as the one shown is Fig. 4.2 with $n = 2$. The parameters of the experimental setup are shown in Table 4.3. It should be noted that the converter is connected to the grid (power supply) via an autotransformer; the autotransformer is used to step down the grid voltage from 230 V to 110 V.

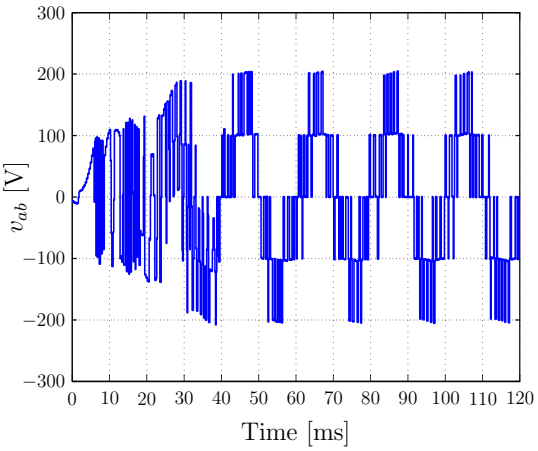
For the performance test the switching constraints are not taken into account in order to highlight the dynamic response of the controller. Thus, a two-step prediction horizon is employed ($N = 2$) so as to keep the computational complexity modest. Furthermore, the weight in the objective function (4.12) is heuristically chosen as $\lambda = 0.2$. Finally, the proportional gain of the PI controllers is chosen as $k_{\text{pr}_1} = k_{\text{pr}_2} = 0.1$, and the integral gain as $k_{\text{int}_1} = k_{\text{int}_2} = 0.7$. The control algorithm was implemented on a dSpace 1104 system with I/O card for real-time control. The schematic diagram of the experimental setup is shown in Fig. 4.11.

4.5.1 Nominal Start-Up

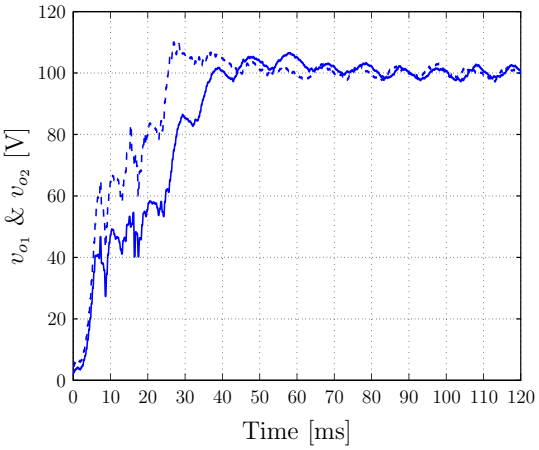
The first case to be examined is that of the dynamic behavior of the CHB rectifier during start-up and nominal conditions. The output voltage reference for both cells is set equal to



(a) Input voltage (dashed line) and current (solid line).



(b) Ac side voltage.



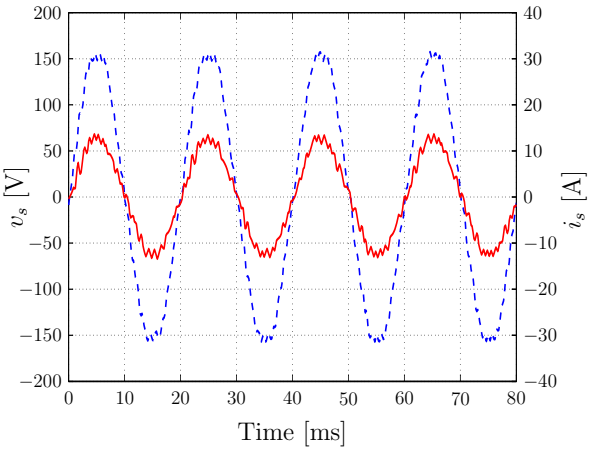
(c) Output voltage of first (solid line) and second cell (dashed line).

Figure 4.12: Experimental results from a single phase rectifier consisting of two cascaded H-Bridge cells for nominal start-up.

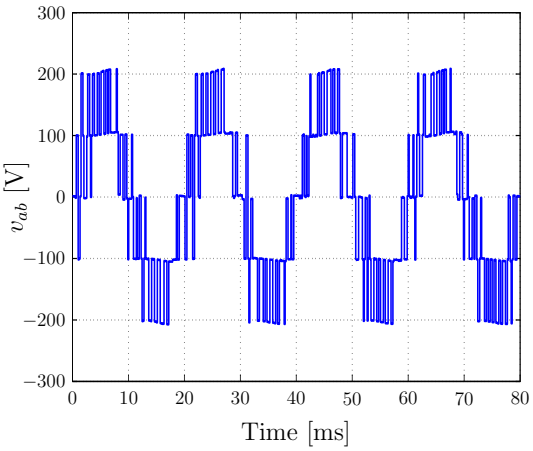
$v_{o,\mathrm{ref}_1} = v_{o,\mathrm{ref}_2} = 100\,\mathrm{V}$. As can be seen in Fig. 4.12, the input current quickly increases in order to charge the capacitors to the demanded voltage levels (Fig. 4.12(a)). After about $t \approx 50\,\mathrm{ms}$ the output voltages of both cells reach their reference values (Fig. 4.12(c)), and the input current reaches its nominal value. Furthermore, the ac side reflected voltage consists of five levels (Fig. 4.12(b)), since the cell voltages are of the same level.

4.5.2 Steady-State Operating Conditions

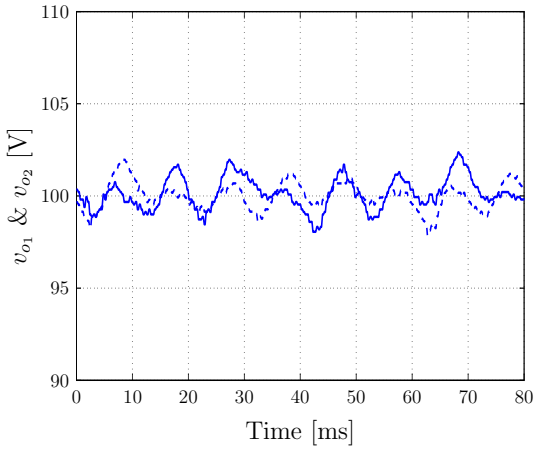
Operating with a switching frequency of about $f_{sw} = 1.1\,\mathrm{kHz}$ at the previously attained operating point, the steady-state performance is examined and the results are presented in Fig. 4.13. The input current i_s is a sinusoidal waveform and in phase with the supply



(a) Input voltage (dashed line) and current (solid line).



(b) Ac side voltage.



(c) Output voltage of first (solid line) and second cell (dashed line).

Figure 4.13: Experimental results under steady-state, nominal operating conditions.

voltage v_s (Fig. 4.13(a)). The harmonic content of the input current is low, resulting in a THD of 3.54%, according to Fig. 4.14 where the current spectrum up to the 41$^{\text{st}}$ harmonic is depicted. It can be observed that the current spectrum is distributed around the 22$^{\text{nd}}$ harmonic, i.e. the most significant harmonics are located in high frequencies corresponding to the switching frequency and the frequencies around it. In Fig. 4.13(b) the five-level reflected voltage to the ac side is illustrated, resulting from the fact that the two-cell converter is operating under balanced output cell voltages (see Fig. 4.13(c)).

4.5.3 Step Change in the Output Reference Voltage

Next, a step change in the reference of the output voltage of the second cell takes place (Fig. 4.15). At time $t \approx 35\,\text{ms}$ the reference is stepped up from $v_{o,\text{ref}_2} = 100\,\text{V}$

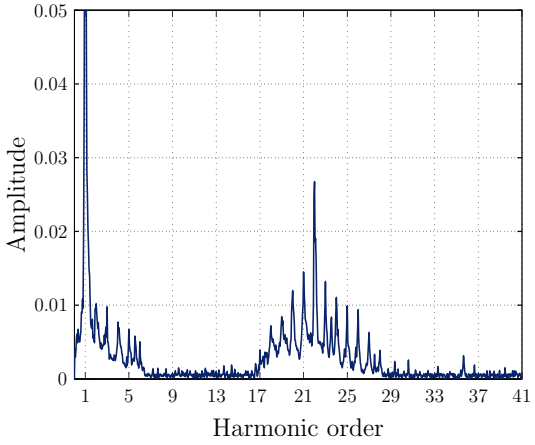


Figure 4.14: Input current spectrum. The THD of the input current i_s is 3.54%. The current is given in p.u..

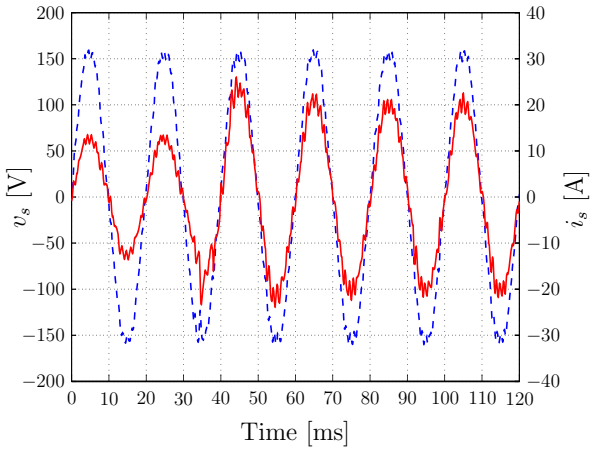
to $v_{o,\text{ref}_2} = 150\,\text{V}$. The output voltage of the second cell reaches its new reference value in about $t \approx 25\,\text{ms}$ without any overshoot or undershoot, while the output voltage of the first cell remains practically unaffected by this change (Fig. 4.15(c)). The input current response to the aforementioned change is depicted in Fig. 4.15(a); the amplitude instantaneously increases, while the unity power factor is maintained. Finally the ac side reflected multilevel voltage (Fig. 4.15(b)) is composed of nine distinctive levels due to the unbalanced output cell voltages, as it is expected.

4.5.4 Load Step

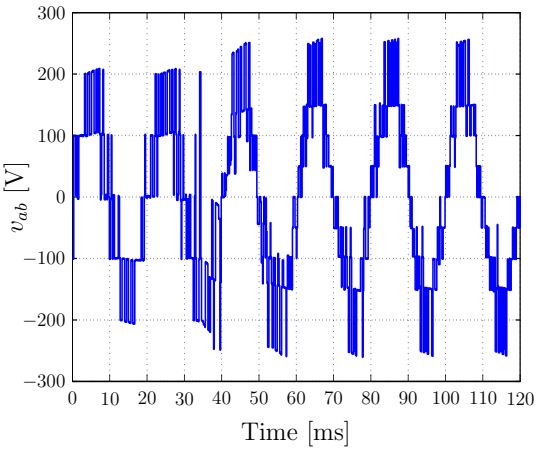
Finally, a step-down change in the load resistance of one cell is examined. For this case the cells operate again at the same voltage potential, i.e. $v_{o,\text{ref}_1} = v_{o,\text{ref}_2} = 100\,\text{V}$, thus a five-level v_{ab} is generated. As shown in Fig. 4.16, at $t \approx 48\,\text{ms}$ the nominal load resistance of the second cell decreases by half, i.e. from $R = 20\,\Omega$ to $R = 10\,\Omega$. The load current of the second cell is instantaneously doubled (Fig. 4.16(d)), while the voltages of both cells remain unaffected by this change, see Fig. 4.16(c).

4.6 Conclusions

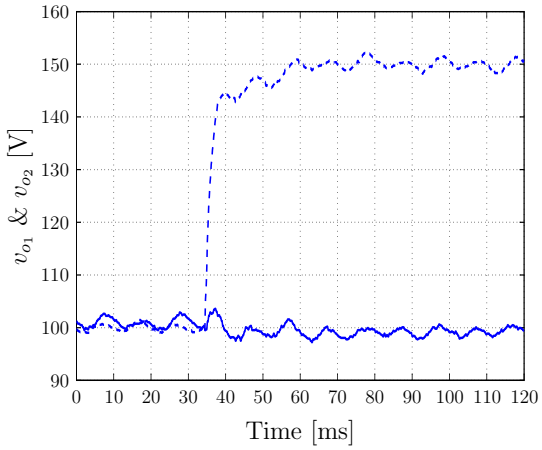
In this chapter, a model predictive control (MPC) approach for the cascaded H-bridge (CHB) multilevel rectifier consisting of n has been presented. In the inner loop, posed in the MPC framework, the input current is regulated to its sinusoidal reference by directly manipulating the switches of the converter. An exhaustive enumeration and search of all the possible switching combinations takes place resulting in a controller which is suitable to predict the behavior of the plant for the entire operation range. Furthermore, in order to maintain the effectiveness of the controller under transient operating conditions and



(a) Input voltage (dashed line) and current (solid line).



(b) Ac side voltage.



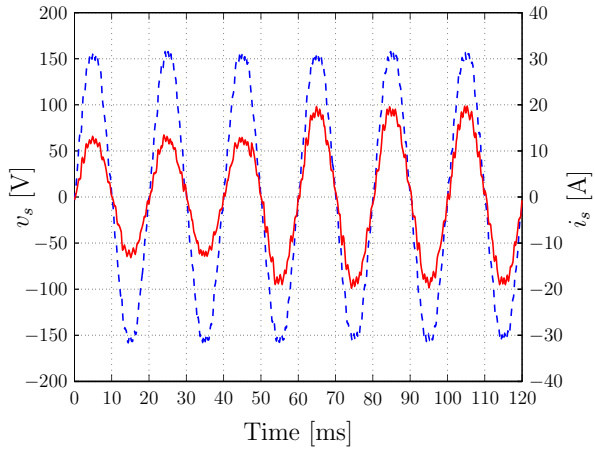
(c) Output voltage of first (solid line) and second cell (dashed line).

Figure 4.15: Experimental results for a step-up change in the output voltage reference of the second cell.

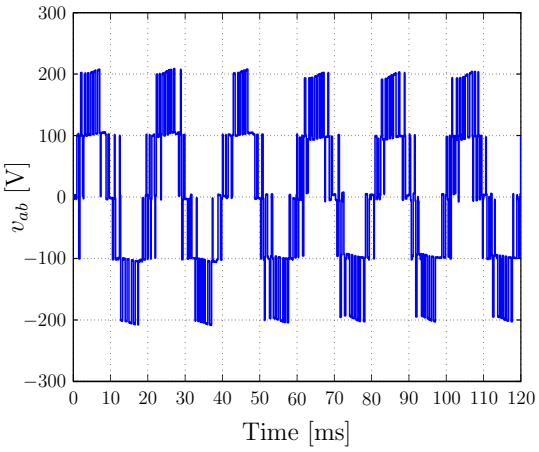
to enhance the dynamic behavior of the system, the deviation of the respective voltages from their references is taken into account. In this way the controller aims to reject all kind of disturbances, including load and output voltage variations.

A key benefit of the proposed algorithm is that despite its design simplicity it is capable of stabilizing the system over the entire operating regime. Furthermore, excessive tuning is avoided, not only due to the fact that the control objectives are expressed in the objective function in a straightforward manner, but also because of the nature of the formulated objective function, wherein only one factor needs to be tuned. Other advantages include the fast dynamics achieved by MPC.

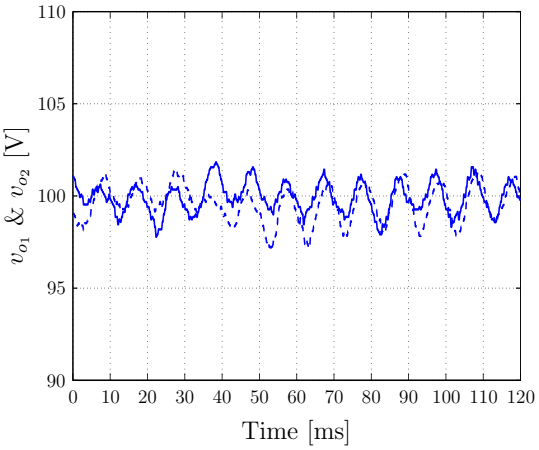
On the other hand the absence of a modulator and the direct manipulation of the



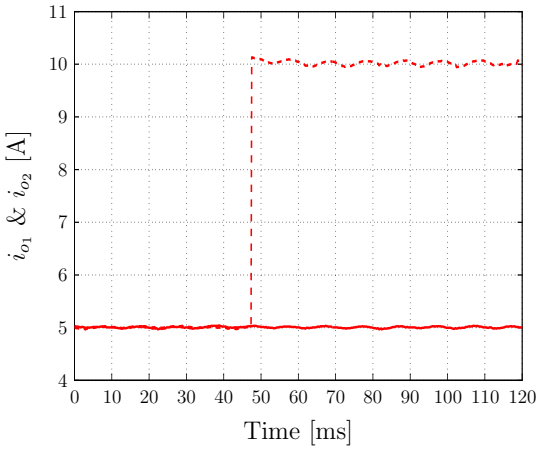
(a) Input voltage (dashed line) and current (solid line).



(b) Ac side voltage.



(c) Output voltage of first (solid line) and second cell (dashed line).



(d) Output current of first (solid line) and second cell (dashed line).

Figure 4.16: Experimental results for a step change in the load of the second cell.

converter switches imply a variable switching frequency. Moreover, the dominant drawback is that the computational power needed increases exponentially when the prediction horizon is extended further into the future. However, methods to significantly reduce the computational effort, e.g. by imposing constraints on the switching transitions, have been proposed. Finally, the performance of the presented control algorithm has been verified by experimental results from a two-cell CHB single-phase multilevel rectifier.

Future Research

Extending the proposed strategy to other topologies, such as the three-phase rectifier, is an interesting topic. An additional research topic would be to find alternative methods to alleviate the computational burden of the proposed algorithm, so as to ease its extension to CHB converters consisting of more cells.

Chapter 5

AC Drives

5.1 Introduction

Adjustable-speed ac drives are used in any application in which there is mechanical equipment powered by motors, such as conveyors, agitators, fans, pumps, blowers and mixers. Dc-ac inverters are used to drive ac electric machines as variable frequency voltage or current sources so that the angular frequency of the rotor shaft can be ramped up and down, or maintained at a required level. Since adjustable-speed ac drives based on voltage source inverters are in constantly increasing demand for numerous industrial applications, there is a continuous need for control schemes that provide, among others, robustness and favorable dynamic behavior. Two of the most common control algorithms are field oriented control (FOC) [73] and direct torque control (DTC) [127].

FOC is a modulation-based approach according to which the electromagnetic torque T_e and the rotor flux $\boldsymbol{\psi}_r$ are controlled indirectly and independently of each other. To do so, a coordinate transformation from stator fixed to a rotor flux oriented coordinate system is employed; to achieve the decoupling between the two controlled variables the rotor flux angle $\delta = \measuredangle \boldsymbol{\psi}_r$ is required[1]. Then the electromagnetic torque is controlled by controlling the q-component of the stator current i_{sq}, while the rotor flux is controlled via the d-component of the stator current i_{sd}. Following, the control signals are derived using conventional PWM methods, such as space vector modulation (SVM). In Fig. 5.1 the block diagram of the FOC strategy is illustrated.

In contrast to that, DTC does not use a modulator; it is rather a direct control approach. The main principle of DTC is to exploit the fast stator flux dynamics so as to directly control the electromagnetic torque T_e and the magnitude of the stator flux

[1]In Section 5.2 the transformation from the three-phase system abc to an orthogonal $dq0$ reference frame is presented in detail.

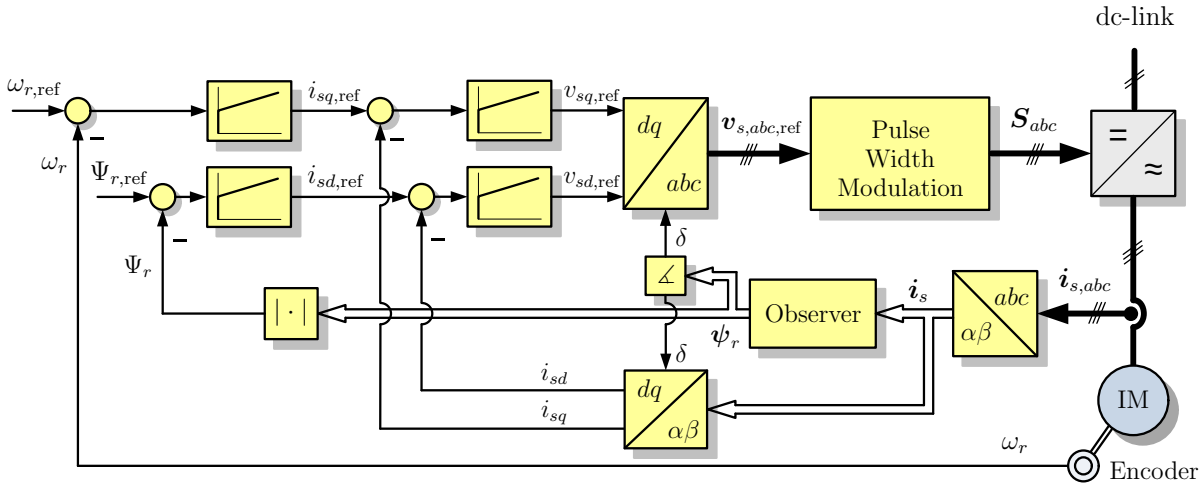


Figure 5.1: Block diagram of field oriented control (FOC).

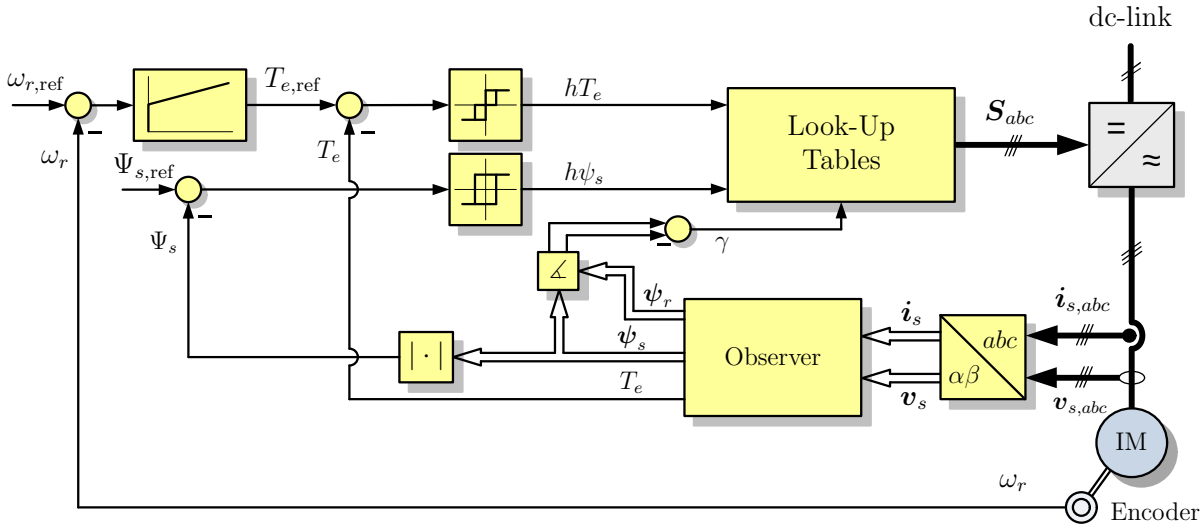


Figure 5.2: Block diagram of direct torque control (DTC).

Ψ_s, by manipulating the stator flux $\boldsymbol{\psi}_s$. In order to achieve this the state of the switches is selected from a look-up table, depending on the stator flux angle and the outputs of hysteresis controllers for flux and torque (see Fig. 5.2 where the block diagram of DTC is depicted).

As it is implied from the absence of a modulator, DTC shows a faster transient response than FOC, while the design procedure is significantly simplified. Furthermore, DTC is efficient over the entire operating range, in contrast to FOC, which is applicable only in a limited range of operation. On the other hand the absence of a modulator has its negative effects: DTC operates with a variable switching frequency, in contrast to the fixed switching frequency of FOC. Moreover, in DTC high current, flux and torque ripples occur, while the switching frequency cannot be directly controlled. Finally, controlling torque and flux at low frequencies becomes a difficult—but not inconceivable—task [21].

Despite the fact that the aforementioned control strategies are considered as two well established methods in three-phase electrical drives control, several challenges, such as design simplicity, high dynamic performance, and reliability, have not been fully addressed. Furthermore, the advent of more fast microprocessors enabled the application of MPC to the field of electrical drive systems [29, 42–44, 48, 70, 102, 109, 112, 124, 126]. In addition, several other MPC based schemes for ac drive systems consisting of a voltage source inverter and an induction machine (IM), or a permanent magnet synchronous machine (PMSM) have been presented in recent years [9, 34, 105, 115, 116].

5.1.1 Model Predictive Control for AC Drives

An very promising MPC-based approach is proposed in [48], and experimentally verified in [112], named model predictive direct torque control (MPDTC). For the regulation of the torque and flux magnitude to their reference values soft constraints, implemented as hysteresis bounds, are present. In addition, since the target applications are the medium voltage (MV) drives, where the switching losses are of dominant importance, the minimization of the switching frequency, as well as low THD of the current, are added to the control objectives. To do so, apart from the hysteresis bounds, the switching transitions are penalized. Finally, in order to emulate a longer prediction horizon the idea of extrapolation is used [41]. Furthermore, the idea is extended from the three-level NPC [48] inverter to the five-level active neutral point clamped inverter (ANPC-5L) in [43]. Finally, the principle of hysteresis bounds and extrapolation are used in [42], with the difference that the flux is indirectly controlled by controlling the stator current instead, according to the proposed method called model predictive direct current control (MPDCC).

In [44] the concept of optimal pulse patterns (OPPs) [19, 113, 114] is adopted, and combined with MPC, resulting in a control strategy named model predictive pulse pattern control (MP^{3}C). Offline computed OPPs that produce minimum THD in the stator windings of the ac machine are used to calculate an optimal stator flux trajectory that the controller tracks in real-time. In this way individual current harmonics of non-triplen order (5^{th}, 7^{th}, 11^{th}, 13^{th}, ...) are reduced and interharmonic components are eliminated [57]. Furthermore, with trajectory tracking control the disadvantages that arise from the use of OPPs in a closed-loop control system, i.e. poor dynamical performance of the drive, and discontinuities in the switching angles that introduce harmonic excursions of the stator currents [58], are overcome [59, 108]. The trajectory tracking is based on MPC. The proposed control scheme is adapted to an ANPC-5L inverter (specifically the ACS 2000 MV drive from ABB) and tested in the lab; the results are presented in [109]. In order to implement the controller and solve the optimal control problem in real-time in a computationally efficient manner an event-based prediction horizon is

employed. Therefore, with the introduced strategy (MP^{3}C) three goals are achieved: (a) fast dynamic control by performing the minimum modifications of the offline-calculated OPPs; (b) a complex observer that tracks the fundamental component of the stator flux or current in real-time is not required; (c) sensitivity of the controller to parameter variation and to measurement noise is reduced. Finally, the inherent voltage redundancies of the ANPC-5L are used in order to control the neutral point potential and the phase capacitor voltages according to the method introduced in [75] (model predictive direct balancing control—MPDBC).

For the low voltage (LV) drives a less computationally demanding MPC approach is proposed. In [29, 102, 124, 126] a dead-beat MPC version, i.e. the length of the prediction horizon is limited to one step $N = 1$, is implemented. The proposed control scheme, called predictive torque control (PTC), offers flexibility in the design process since the optimization problem captures the control objectives, i.e. the regulation of the torque and flux magnitude to their reference values, in a straightforward manner. In addition, constraints like current limitation can be explicitly imposed. Furthermore, the design simplicity of PTC makes its implementation for different types of inverters a straightforward procedure. Thereby, PTC is implemented for the two-level inverter [102], for the three-level NPC inverter [124], for the FC inverter [126], and for the CHB inverter [29].

However, the torque and current ripples that PTC delivers are still very high compared to modulator-based strategies, such as FOC, with the same sampling time. In the medium- and high-voltage (HV) drives field these large ripples are less important than a low switching frequency of the inverter. The higher the power range of a drive system is, the more dominant are the switching losses in relation to the total losses. For LV drive systems, though, the switching losses are less important than a good quality of the torque and the current. By achieving a higher switching frequency these drawbacks can be overcome. Nonetheless, in order to reach a higher switching frequency which is in the range of FOC the sampling interval has to be decreased; this leads to higher hardware requirements and costs for the whole drive system. A detailed comparison of FOC, DTC and PTC can be found in [74] and [120].

A solution to reduce the torque ripple is to allow the switchings to take place in between the sampling interval. In this way the ripple can be decreased without the need of a further reduction of the sampling interval. However, the maximum switching frequency is still limited to half the sampling frequency. In [84] a method to calculate an optimal switching time point for PTC and a PMSM is presented.

Motivated by the lack of a control strategy that targets LV drives and achieves reduction of torque and current ripples, without a subsequent increase in the sampling frequency—and thus in the cost of the hardware—we propose an algorithm for the determination of a variable switching time point for PTC of IMs. An optimization problem

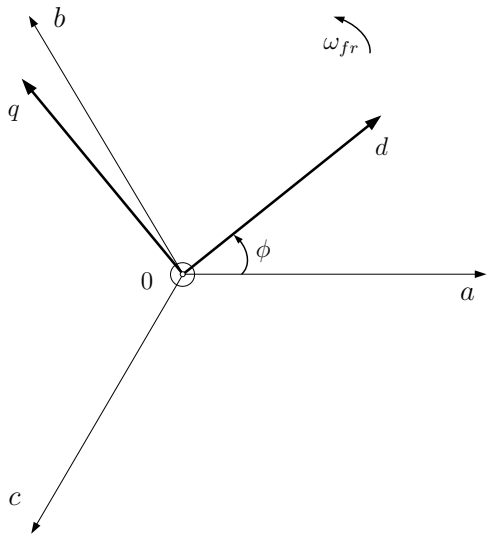


Figure 5.3: Rotating $dq0$ reference frame.

is formulated and solved in real-time that takes into account the torque ripple. Thereby, apart from the regulation of the torque and the flux magnitude to their references, an additional control objective should be met: the minimization of the torque ripple. The result of the optimization procedure is a variable in time point at which the switches of the inverter change state; at this time point the derived optimal control signals (i.e. the control input) are applied to the inverter.

5.2 The $dq0$ Reference Frame

In the field of electrical engineering it is usual to employ the $dq0$ mathematical transformation in order to simplify the analysis of three-phase circuits. With the help of the $dq0$ transformation the quantities of three-phase abc systems are transformed into an orthogonal $dq0$ reference frame, having a direct (d), a quadrature (q), and a zero (0) axis, which can be either rotating with an angular speed ω_{fr}, or stationary [79].

By utilizing the $dq0$ transformation, a variable $\boldsymbol{\chi}_{abc} = [\chi_a \ \chi_b \ \chi_c]^T$ in the three-phase system (abc) can be transformed to $\boldsymbol{\chi}_{dq0} = [\chi_d \ \chi_q \ \chi_0]^T$ in an equivalent but linearly independent $dq0$ coordinate system through

$$\boldsymbol{\chi}_{dq0} = \boldsymbol{K}(\phi)\boldsymbol{\chi}_{abc}\,, \tag{5.1}$$

where ϕ is the angle between the a-axis of the three-phase abc system and the d-axis of the $dq0$ reference frame (see Fig. 5.3). The matrix $\boldsymbol{K}(\phi)$ is given by

$$\boldsymbol{K}(\phi) = \frac{2}{3}\begin{bmatrix} \cos\phi & \cos(\phi - \frac{2\pi}{3}) & \cos(\phi + \frac{2\pi}{3}) \\ -\sin\phi & -\sin(\phi - \frac{2\pi}{3}) & -\sin(\phi + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}. \tag{5.2}$$

For a frame rotating with an angular speed ω_{fr}, as in Fig. 5.3, then $\phi = \omega_{fr}t + \phi_0$. On the other hand, if the frame is stationary, then ϕ is time invariant. For the special case when $\phi = 0$, then the reference frame is fixed and the d-axis is aligned with the a-axis; the transformation matrix $\boldsymbol{K}$ is

$$\boldsymbol{K} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}. \tag{5.3}$$

Note that when $\phi = 0$, the $dq0$ transformation is also called $\alpha\beta$ transformation, thus $\boldsymbol{\chi}_{\alpha\beta 0} = \boldsymbol{K}\boldsymbol{\chi}_{abc}$, with $\boldsymbol{\chi_{\alpha\beta 0}} = [\chi_\alpha \ \chi_\beta \ \chi_0]^T$.

Finally, for a balanced three-phase system, i.e. $\chi_a + \chi_b + \chi_c = 0$, the zero component of $\boldsymbol{\chi}_{dq0}$ is zero, thus the transformation can be written as (for a rotating frame)

$$\boldsymbol{\chi}_{dq} = \begin{bmatrix} \chi_d & \chi_q \end{bmatrix}^T = \boldsymbol{K}(\phi)\boldsymbol{\chi}_{abc}\,, \tag{5.4}$$

with

$$\boldsymbol{K}(\phi) = \frac{2}{3}\begin{bmatrix} \cos\phi & \cos(\phi - \frac{2\pi}{3}) & \cos(\phi + \frac{2\pi}{3}) \\ -\sin\phi & -\sin(\phi - \frac{2\pi}{3}) & -\sin(\phi + \frac{2\pi}{3}) \end{bmatrix}, \tag{5.5}$$

or for a stationary frame with $\phi = 0$

$$\boldsymbol{\chi}_{\alpha\beta} = \begin{bmatrix} \chi_\alpha & \chi_\beta \end{bmatrix}^T = \boldsymbol{K}(\phi)\boldsymbol{\chi}_{abc}\,, \tag{5.6}$$

with

$$\boldsymbol{K} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}. \tag{5.7}$$

5.3 Physical System

5.3.1 Continuous-Time Model

5.3.1.1 Two-Level Inverter

Fig. 5.4 shows the three-phase inverter connected to an IM. The two IGBTs in each one of the three phases are complementary, i.e. if the upper switch is *off*, the lower switch has to be turned *on* and vice versa. In each phase the inverter produces voltages $-\frac{V_{\mathrm{dc}}}{2}$, and $\frac{V_{\mathrm{dc}}}{2}$, where V_{dc} is the dc-link voltage. This leads to $2^3 = 8$ different switching possibilities; modeling the switching state with the variables $u_a, u_b, u_c \in \{0, 1\}$, where "1" corresponds to the case of the positive phase voltage, and "0" to the case of the negative voltage, eight possible combinations of the switching states $\boldsymbol{u}_{abc} = [u_a \ u_b \ u_c]^T$ are produced.

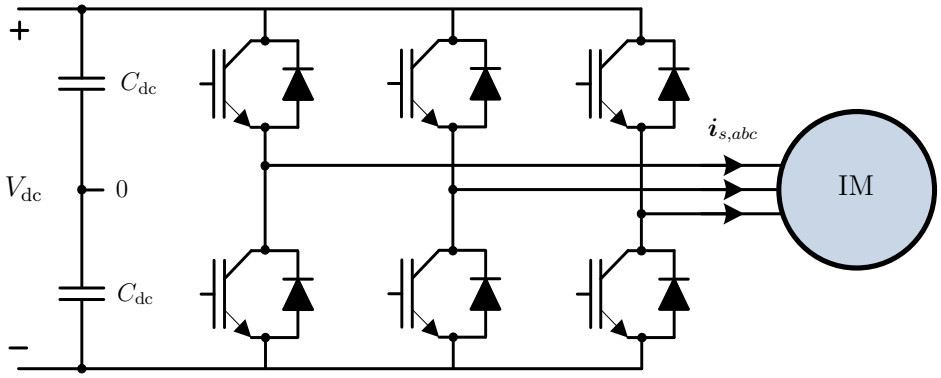


Figure 5.4: Two-level voltage source inverter driving an IM.

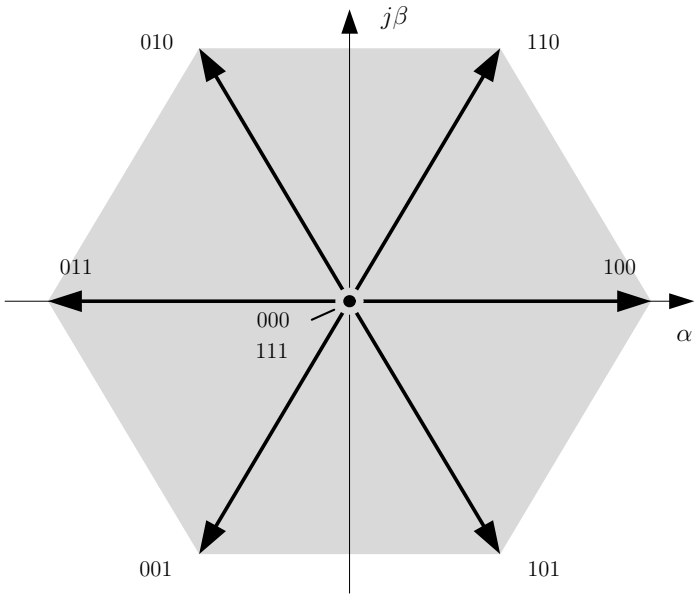


Figure 5.5: Voltage vectors on the $\alpha\beta$ plane of a two-level voltage source inverter and the corresponding switching states.

Using the transformation matrix given by (5.7) the actual voltages applied to the machine terminals are transformed from the *abc* plane to the $\alpha\beta$ plane resulting in seven unique voltage vectors $\boldsymbol{v}_{\alpha\beta}$ (Fig. 5.5) given by

$$\boldsymbol{v}_{\alpha\beta} = V_{\mathrm{dc}} \boldsymbol{K} \boldsymbol{u}_{abc} \,. \tag{5.8}$$

Finally, the switching states $[0\ 0\ 0]^T$ and $[1\ 1\ 1]^T$ are called zero switching states and produce zero voltage vectors, and short-circuit the machine terminals. The others are the active switching states that produce active voltage vectors, and they are the long vectors shown in Fig. 5.5 that form the outer hexagon.

5.3.1.2 Three-Level Inverter

The three-level NPC voltage source inverter [106] driving an IM is shown in Fig. 5.6. The voltage of the dc-link is split by the capacitors C_{dc_1} and C_{dc_2} into three levels. Using the common terminal point n as reference, the inverter produces at each phase voltages $v_{\mathrm{ph},x} \in \{-\frac{V_{\mathrm{dc}}}{2}, 0, \frac{V_{\mathrm{dc}}}{2}\}$, where $\{x \in \mathcal{X} \mid \mathcal{X} = \{a, b, c\}\}$, and V_{dc} is the dc-link voltage. By introducing the integer variables $u_a, u_b, u_c \in \{-1, 0, 1\}$ in order to model the switching state in each phase leg, i.e. the values $-1, 0, 1$ correspond to the phase voltages $-\frac{V_{\mathrm{dc}}}{2}, 0, \frac{V_{\mathrm{dc}}}{2}$, respectively, $3^3 = 27$ possible combinations of the switching states $\boldsymbol{u}_{abc} = [u_a\ u_b\ u_c]^T$ are produced (see Fig 5.7—positive switching state denoted with $+$, negative one with $-$).

The actual voltages applied to the machine terminals are transformed from the *abc* plane to the $\alpha\beta$ plane resulting in 19 unique voltage vectors (Fig 5.7). For the case of the three-level NPC inverter, the switching states $[-1\ -1\ -1]^T$, $[0\ 0\ 0]^T$ and $[1\ 1\ 1]^T$ are called zero switching states and produce zero voltage vectors, and short-circuit the machine terminals. The others are the active switching states that produce active voltage vectors, and they are divided in three groups. The first group consists of the six long vectors that form the outer hexagon, the second group includes six vectors of medium length, and the third group twelve short vectors that form the inner hexagon. The voltage vectors $\boldsymbol{v}_{\alpha\beta}$ are given by

$$\boldsymbol{v}_{\alpha\beta} = \frac{V_{\mathrm{dc}}}{2} \boldsymbol{K} \boldsymbol{u}_{abc} \,. \tag{5.9}$$

where $\boldsymbol{K}$ is transformation matrix (5.7).

The neutral point potential v_n depends on the state of charge of the two dc-link capacitors. This is affected only when a current is flowing through the capacitors, a case which is true when a switch position is zero. The rate of change of the neutral point voltage is given by

$$\frac{dv_n}{dt} = -\frac{1}{2C_{\mathrm{dc}}} \sum_{x \in \mathcal{X}} (1 - |u_x|) i_{sx} \,, \tag{5.10}$$

where i_{sx} is the stator current of phase x, and C_{dc} is the capacitance of the dc-link capacitors. Assuming a balanced load, i.e.

$$i_{sa} + i_{sb} + i_{sc} = 0 \,, \tag{5.11}$$

then (5.10) can be written as

$$\frac{dv_n}{dt} = \frac{1}{2C_{\mathrm{dc}}} |\boldsymbol{u}_{abc}|^T \boldsymbol{K}^{-1} \boldsymbol{i}_{s\alpha\beta} \,, \tag{5.12}$$

where $\boldsymbol{i}_{s\alpha\beta}$ is the stator current in the stator $\alpha\beta$ reference frame. Furthermore, $|\boldsymbol{u}_{abc}| = [|u_a|\ |u_b|\ |u_c|]^T$ is the componentwise absolute values of the switching state in each phase leg. For more details about the nature of the neutral point potential and existing balancing strategies, the reader is referred to [24, 57, 100].

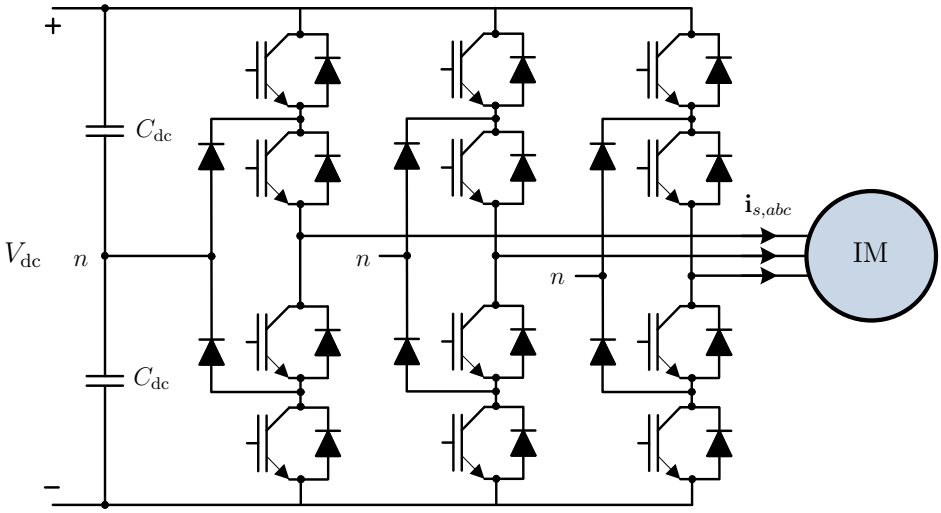


Figure 5.6: Three-level neutral point clamped (NPC) voltage source inverter driving an induction machine (IM).

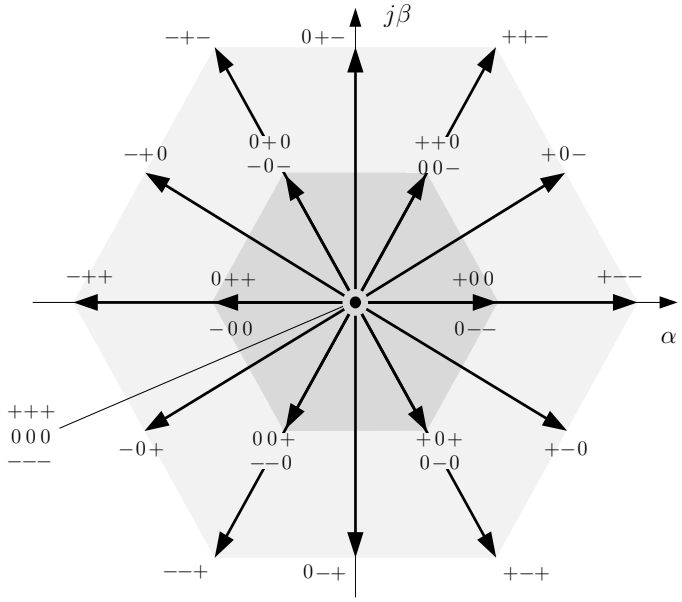


Figure 5.7: Voltage vectors on the $\alpha\beta$ plane of a three-level NPC voltage source inverter and the corresponding switching states.

5.3.1.3 Induction Machine

In order to derive a mathematical model appropriate for the controller, the dynamics of the IM are modeled in the stator $\alpha\beta$ reference frame. As state variables are considered the stator current $\boldsymbol{i}_s$, the stator flux $\boldsymbol{\psi}_s$, and the rotor rotational speed ω_r. The continuous-

time state equations are [56]

$$\tau_{sr}\frac{d\boldsymbol{i}_s}{dt} + \boldsymbol{i}_s = j\omega_r\tau_{sr}\boldsymbol{i}_s + \frac{1}{r_{sr}}\Big(\frac{1}{\tau_r} - j\omega_r\Big)\boldsymbol{\psi}_s + \frac{1}{r_{sr}}\boldsymbol{v}_s \tag{5.13a}$$

$$\frac{d\boldsymbol{\psi}_s}{dt} = -r_s\boldsymbol{i}_s + \boldsymbol{v}_s \tag{5.13b}$$

$$\frac{d\omega_r}{dt} = \frac{p}{H}\big(T_e - T_\ell\big) \tag{5.13c}$$

where the stator voltage $\boldsymbol{v}_s$ is in the stator $\alpha\beta$ reference plane. Based on the model parameters, i.e. the stator r_s and the rotor r_r resistances, and the stator l_s, the rotor l_r and the mutual l_m inductances, the coefficients in (5.13) are given as $\tau_r = l_r/r_r, \tau_{sr} = \sigma l_s/r_{sr}$, and $r_{sr} = r_s + l_s/l_r \cdot r_r$, with $\sigma = 1 - l_m^2/(l_s l_r)$. Variable H stands for the inertia, T_ℓ for the mechanical load torque, and the electromagnetic torque T_e is given by

$$T_e = \frac{3}{2}p\big(\boldsymbol{\psi}_s \times \boldsymbol{i}_s\big)\,, \tag{5.14}$$

where p is the number of pole pairs.

Finally, the magnitude of the stator flux is given by

$$\Psi_s = \sqrt{\psi_{s\alpha}^2 + \psi_{s\beta}^2}\,. \tag{5.15}$$

5.3.2 Discrete-Time Model

5.3.2.1 Two-Level Inverter—Induction Machine System

The discrete-time model of the two-level inverter—IM system which can be used as prediction model for MPC, is derived by discretizing (5.13) and (5.14) using the forward Euler approximation. The discrete-time state-space model of the plant is of the form

$$\boldsymbol{x}(k+1) = (\boldsymbol{I} + \boldsymbol{A}T_s)\boldsymbol{x}(k) + \boldsymbol{B}T_s\boldsymbol{u}(k) \tag{5.16a}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\big(\boldsymbol{x}(k)\big)\,, \tag{5.16b}$$

where the state vector is selected to be

$$\boldsymbol{x} = \begin{bmatrix} i_{s\alpha} & i_{s\beta} & \psi_{s\alpha} & \psi_{s\beta} \end{bmatrix}^T, \tag{5.17}$$

the switching states $\boldsymbol{u}_{abc}$ serve as the input vector, and the output vector is

$$\boldsymbol{y} = \begin{bmatrix} T_e & \Psi_s \end{bmatrix}^T. \tag{5.18}$$

The matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and the state-dependent matrix $\boldsymbol{C}$ are the matrices of the continuous-time model given by

$$\boldsymbol{A} = \begin{bmatrix} -\frac{1}{\tau_{sr}} & -\omega_r & \frac{1}{\gamma\tau_r} & \frac{\omega_r}{\gamma} \\ \omega_r & -\frac{1}{\tau_{sr}} & -\frac{\omega_r}{\gamma} & \frac{1}{\gamma\tau_r} \\ -r_s & 0 & 0 & 0 \\ 0 & -r_s & 0 & 0 \end{bmatrix}, \tag{5.19}$$

$$\boldsymbol{B} = V_{\mathrm{dc}} \begin{bmatrix} \frac{1}{\gamma} & 0 \\ 0 & \frac{1}{\gamma} \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \boldsymbol{K}, \tag{5.20}$$

$$\boldsymbol{C} = \begin{bmatrix} \frac{3}{2}p\big(x_3(k)x_2(k) - x_4(k)x_1(k)\big) \\ \sqrt{x_3^2(k) + x_4^2(k)} \end{bmatrix}, \tag{5.21}$$

with $\gamma = \tau_{sr} r_{sr}$. Finally, $\boldsymbol{I}$ is the identity matrix and T_s is the sampling interval.

5.3.2.2 Three-Level Inverter—Induction Machine System

Following the same procedure as in Section 5.3.2.1 the discrete-time model of the three-level NPC inverter—IM system is derived. The state vector includes the stator current and flux in the $\alpha\beta$ frame, as well as the state of the inverter, i.e. the neutral point potential. Hence, the state vector is

$$\boldsymbol{x} = \begin{bmatrix} i_{s\alpha} & i_{s\beta} & \psi_{s\alpha} & \psi_{s\beta} & v_n \end{bmatrix}^T, \tag{5.22}$$

Moreover, the switching states $\boldsymbol{u}_{abc}$ serve as the input vector, and the output vector is

$$\boldsymbol{y} = \begin{bmatrix} T_e & \Psi_s & v_n \end{bmatrix}^T. \tag{5.23}$$

Therefore, by discretizing (5.12), (5.13) and (5.14) using the forward Euler approximation the state-space model of the drive is

$$\boldsymbol{x}(k+1) = (\boldsymbol{I} + \boldsymbol{A}T_s)\boldsymbol{x}(k) + \boldsymbol{B}_1 T_s \boldsymbol{u}(k) + \boldsymbol{B}_2\big(\boldsymbol{x}(k)\big) T_s |\boldsymbol{u}(k)| \tag{5.24a}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\big(\boldsymbol{x}(k)\big). \tag{5.24b}$$

The continuous-time matrices $\boldsymbol{A}$, $\boldsymbol{B}_1$, $\boldsymbol{B}_2$ and $\boldsymbol{C}$ (matrices $\boldsymbol{B}_2$ and $\boldsymbol{C}$ are state-dependent) are given by

$$\boldsymbol{A} = \begin{bmatrix} -\frac{1}{\tau_{sr}} & -\omega_r & \frac{1}{\gamma\tau_r} & \frac{\omega_r}{\gamma} & 0 \\ \omega_r & -\frac{1}{\tau_{sr}} & -\frac{\omega_r}{\gamma} & \frac{1}{\gamma\tau_r} & 0 \\ -r_s & 0 & 0 & 0 & 0 \\ 0 & -r_s & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{5.25}$$

$$\boldsymbol{B}_1 = \frac{V_{\mathrm{dc}}}{2} \begin{bmatrix} \frac{1}{\gamma} & 0 \\ 0 & \frac{1}{\gamma} \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \boldsymbol{K}, \tag{5.26}$$

$$\boldsymbol{B}_2 = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{D} \end{bmatrix}, \tag{5.27}$$

$$\boldsymbol{C} = \begin{bmatrix} \frac{3}{2}p\big(x_3(k)x_2(k) - x_4(k)x_1(k)\big) \\ \sqrt{x_3^2(k) + x_4^2(k)} \\ x_5(k) \end{bmatrix}. \tag{5.28}$$

In (5.27) $\mathbf{0}$ is a 4×3 zero matrix, and the matrix $\boldsymbol{D}$ is

$$\boldsymbol{D} = \frac{1}{2C_{\mathrm{dc}}} \boldsymbol{x}^T(k) \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \boldsymbol{K}^{-T}. \tag{5.29}$$

5.4 Variable Switching Point Predictive Torque Control (VSP^{2}TC)

In this work MPC is used to control the output variables, i.e. the torque and the magnitude of the stator flux. Furthermore, an additional control objective is the reduction of the torque ripple. Based on PTC introduced in [29] and [102] the proposed algorithm aims to meet both control objectives without a significant increase of the switching frequency. This is achieved by solving an optimization problem in real-time; a variable switching point is calculated within the prediction horizon of a fixed length based on the minimization of the torque ripple.

5.4.1 Control Problem

In direct switching strategies such as DTC and PTC high torque ripples occur, since a switching state is applied for at least one sampling interval T_s. An active switching state leads to higher current and torque ripples compared to the zero switching state. Hence, if it could be applied for a time period t less than one sampling interval $t < T_s$, then the torque ripple could be reduced.

This principle is shown in Fig. 5.8, where for reasons of simplicity only two switching states are assumed to be applied: an active switching state resulting in a high positive torque slope, and a zero switching state resulting in a low negative slope. As can be seen, the active switching state leads to a high ripple; the longer it is applied, the higher the ripple. If the switching can take place *in between* the sampling interval, the switching state

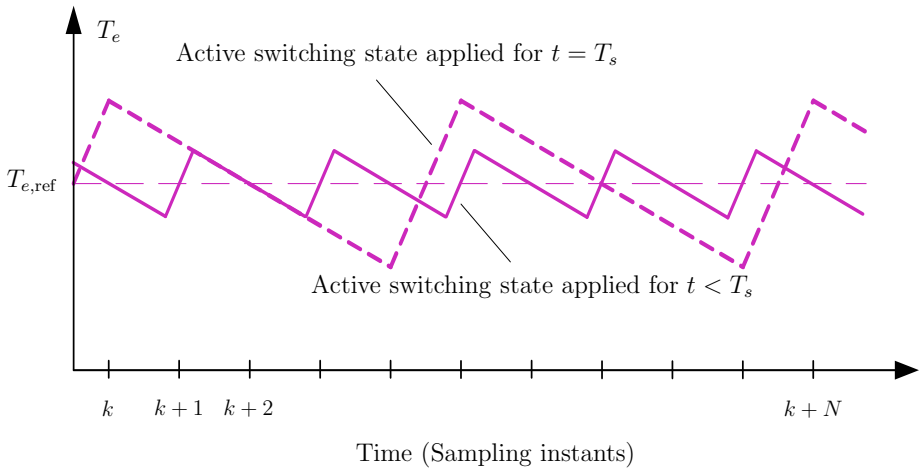


Figure 5.8: Principle of the proposed strategy for torque ripple reduction.

that results in the high ripple would be applied for less time; in this way the ripple could be reduced. However, as it can be observed in Fig. 5.8, this leads to a higher switching frequency. It should be mentioned, though, that this higher switching frequency can be achieved with the same sampling interval T_s. Furthermore, for both cases, the maximum switching frequency is limited to half the sampling frequency, as each IGBT can switch only once during one T_s.

5.4.2 Control Algorithm

5.4.2.1 VSP²TC for the Two-Level Inverter

In this section the derivation of the proposed algorithm (VSP²TC) for the two-level inverter (see Fig. 5.4) is presented in detail.

By selecting a switching state at a time instant within the sampling interval, i.e. $kT_s \leq (k + n_{int}^{(k)})T_s \leq (k+1)T_s$, where the superscript k denotes the k^{th} interval, and $n_{int}^{(k)} \in [0, 1]$, rather than at the beginning of each interval $\big(kT_s, (k+1)T_s, \ldots, (k+N)T_s\big)$, with $N \in \mathbb{N}^+$, the goal is to reduce the torque ripple. The procedure for the calculation of the variable switching point comprises the following steps, executed at step k.

Step 1: At step k the switching state calculated at the previous sampling interval $\boldsymbol{u}(k - 1 + n_{int}^{(k-1)})$ is applied.

Step 2: According to (5.16b), the electromagnetic torque at time instant kT_s is calculated from the measurements of the stator currents and the rotor speed. Using (5.16a), and by applying the same switching state $\boldsymbol{u}(k) = \boldsymbol{u}(k - 1 + n_{int}^{(k-1)})$ for the entire sampling interval T_s, the predicted values of the stator current and flux are calculated; based on these values the torque at step $k + 1$ is computed (see Fig. 5.9). Considering a constant torque slope m for one T_s, the following affine (linear plus offset) expression describes the

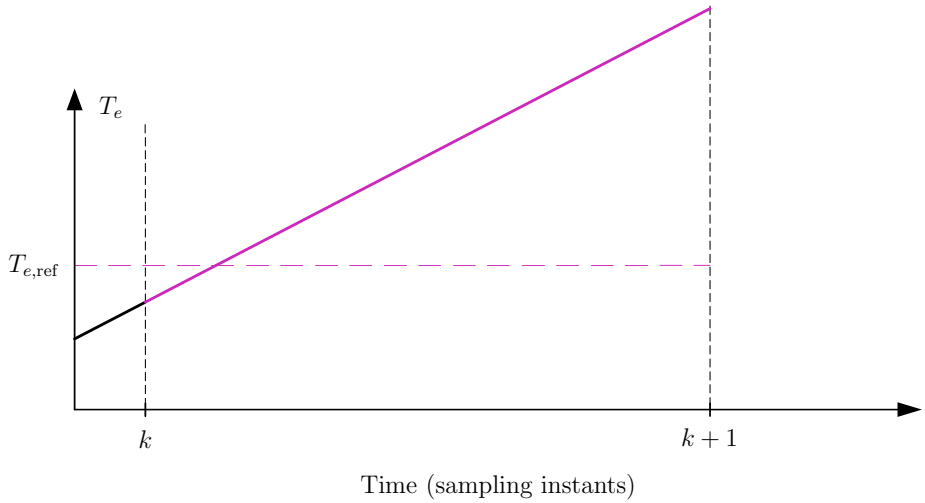


Figure 5.9: A variable switching point is calculated in order to minimize the torque ripple; the calculation of the torque T_e at step $k+1$ is shown.

relationship between the torque at step k and at $k+1$

$$T_e(k+1) = T_e(k) + m(k)T_s\,. \tag{5.30}$$

Step 3: The predicted state and output variables are recomputed assuming that the switching state at step k can be anyone out of the eight possible. The corresponding torque slopes are calculated according to the modified (5.30)

$$T_{e_z}(k+1) = T_e(k) + m_z(k)T_s\,, \tag{5.31}$$

with $z \in \{0, 1, \ldots, 7\}$ denoting the selected switching state. Since $T_s \ll T_1$, where T_1 is the fundamental period, the torque slopes can be considered as, without loss of generality, to remain the same for the successive sampling interval, i.e. $m_z(k) = m_z(k+1)$, see Fig. 5.10.

Step 4: Setting as goal the torque to reach its reference value at step $k+1$, the variable switching point $n_{int}^{(k)}$ is calculated (Fig. 5.11): the intersection between the trajectory of the actual torque $T_e(k)$ and each one of the *possible* trajectories of the calculated torque $T_{e_z}(k+1)$ is computed; its projection onto the time axis equals to the variable switching point (Fig. 5.12). Hence, the variable switching point is given by:

$$t_z = \frac{T_{e,\text{ref}} - T_e(k) - m_z(k+1)T_s}{m(k) - m_z(k+1)}\,, \tag{5.32}$$

where t_z is the projection of the intersection point T_{e,int_z}, resulting from the z switching state, onto the x-axis, i.e. $t_z = n_{int_z}^{(k)} T_s$. Thus, the torque at the intermediate step is $T_{e,int_z} = T_{e_z}(k + n_{int_z}^{(k)})$. Here, it should be pointed out that the switching point can, theoretically, be in the range $t_z \in (-\infty, +\infty)$. However, it is limited to the range $t_z \in [0, T_s]$,

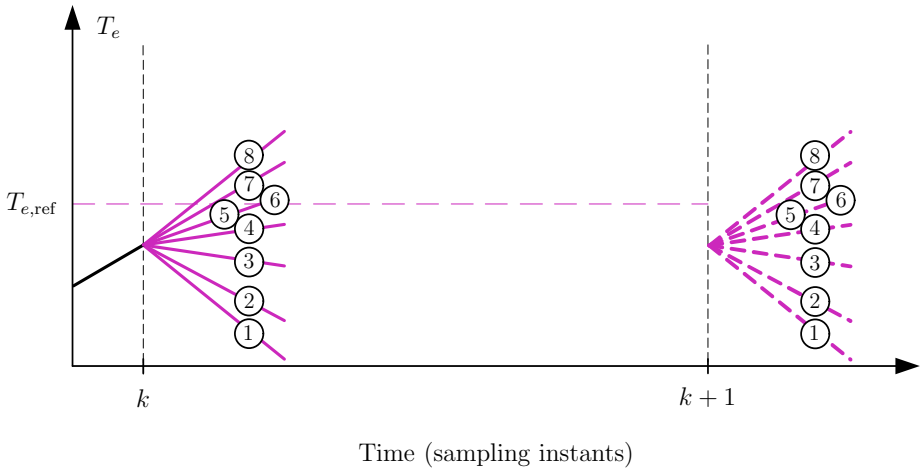


Figure 5.10: A variable switching point is calculated in order to minimize the torque ripple; the calculation of all possible torque slopes is shown. The possible torque trajectories are calculated at step k; the resulting seven slopes are considered to be the same for the successive sampling interval.

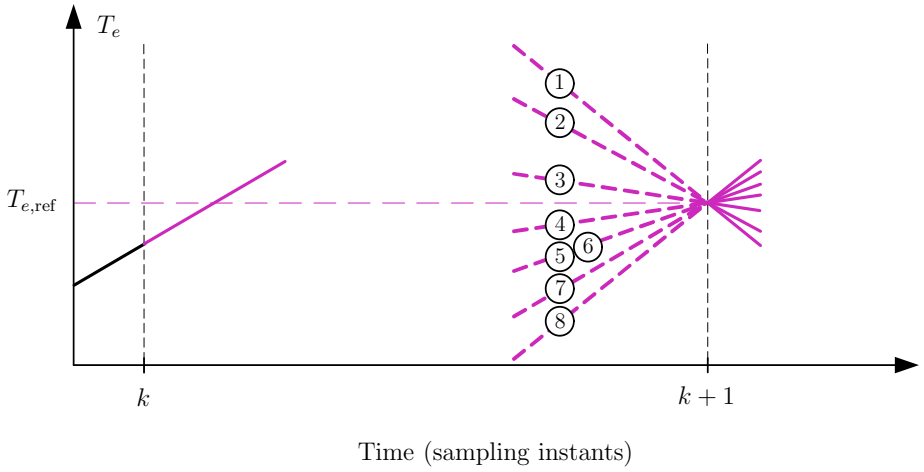


Figure 5.11: A variable switching point is calculated in order to minimize the torque ripple; the possible torque trajectories to which $T_{e_z}(k+1) = T_{e,\text{ref}}$ applies are shown. The *actual* torque slope m at k is calculated by applying the same switching state $\boldsymbol{u}(k) = \boldsymbol{u}(k-1+n_{int}^{(k-1)})$ for the entire T_s. The goal is to get $T_e(k+1) = T_{e,\text{ref}}$. To do so, the torque trajectory must be one of the seven unique trajectories.

with "0" corresponding to PTC.

Step 5: By taking into account the variable switching point, the predicted values of the state and output variables are calculated at step $k + n_{int_z}^{(k)}$. This means that in (5.16) t_z is used instead of T_s. Following, the predictions of the variables of interest are computed for each switching state, in a similar manner as before, with the difference that now the corresponding time interval $T_s - t_z$ is used instead of t_z.

Step 6: In a last step an objective function is formulated and it is minimized in real-time.

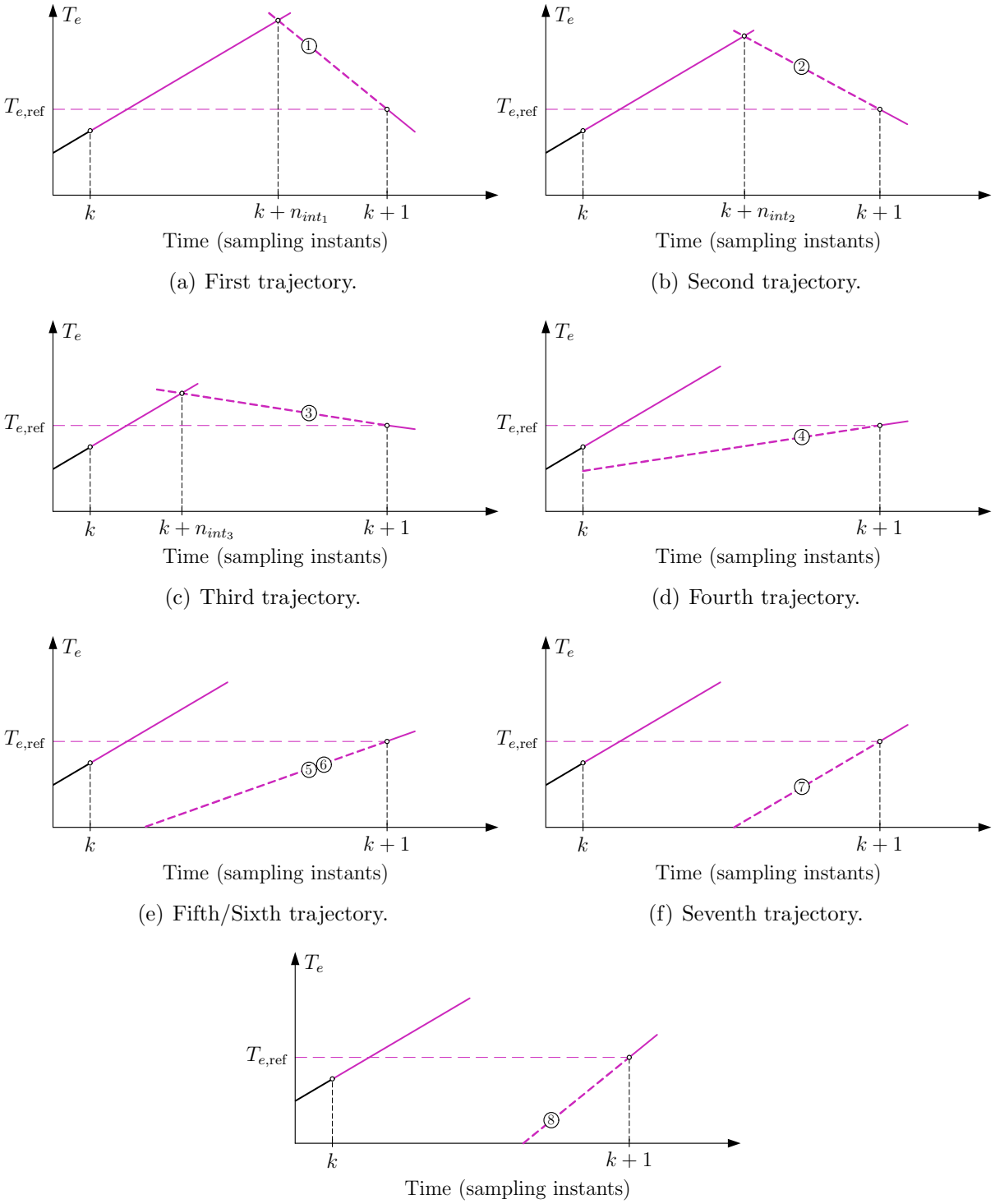


Figure 5.12: A variable switching point is calculated in order to minimize the torque ripple; the calculation of the variable switching point is shown. The point where the torques of steps k and $k + 1$ intersect is calculated. The seven candidate torque trajectories are shown; torques (1), (2) and (3) at $k + 1$ intersect with $T_e(k)$, while torque trajectories (4)—(8) does not.

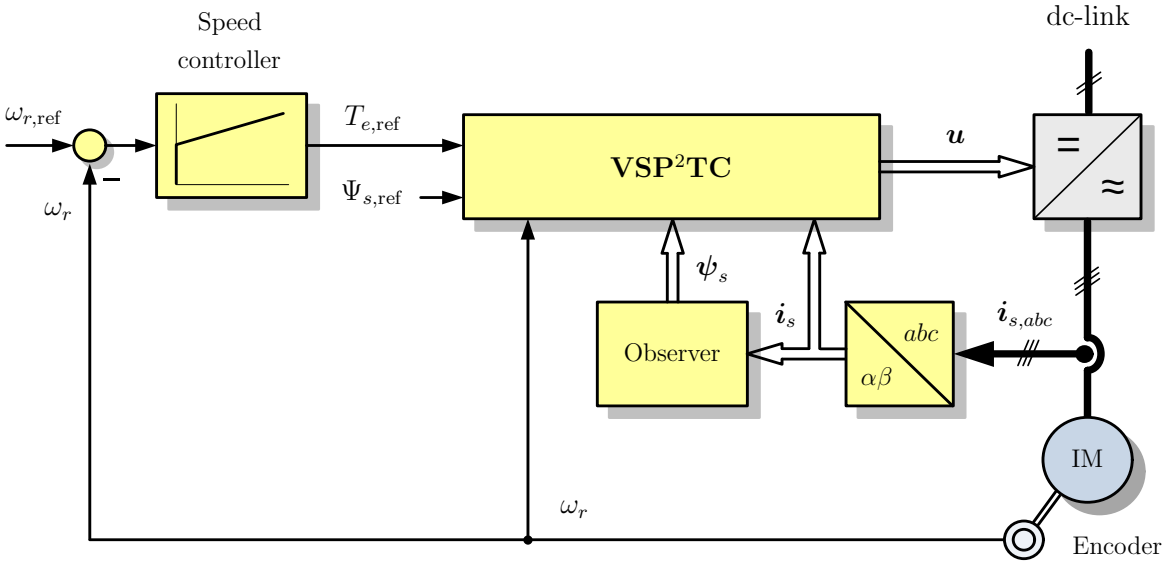


Figure 5.13: Block diagram of the variable switching point predictive torque controller (VSP^{2}TC) for the two-level voltage source inverter driving an IM.

The chosen function is:

$$J(k) = \sum_{\xi \in \mathcal{S}} \left(||T_{e,\text{ref}} - T_e(k+\xi|k)||_2^2 + \lambda ||\Psi_{s,\text{ref}} - \Psi_s(k+\xi|k)||_2^2 \right). \tag{5.33}$$

In (5.33) the squared 2-norm is chosen in order to penalize more heavily the deviations from the reference values. Furthermore, the set $\mathcal{S}$ is defined as $\mathcal{S} = \{n_{int}, 1\}$. Finally, the weighting factor $\lambda > 0$ sets the trade-off between the electromagnetic torque error and the stator flux magnitude error.

Subsequently, by taking into account the objective function (5.33) and the system dynamics (5.16) an optimization problem is formulated:

$$\begin{aligned} &\text{minimize} && J(k) \\ &\text{subject to} && (5.16)\,. \end{aligned} \tag{5.34}$$

The underlying optimization problem is solved in real-time every T_s. The switching state $\boldsymbol{u}(k)$ that results in the minimum associated cost is considered to be the optimal solution, i.e. $\boldsymbol{u}^*(k)$, and it is applied to the inverter at time instant $(k + n_{int}^{(k)})T_s$.

Finally, at the next time-step, the whole procedure is repeated with new measurements or estimates. In Fig. 5.13 the block diagram that consists of the outer speed and the inner loop, constituted by VSP^{2}TC, is shown.

5.4.2.2 VSP^{2}TC for the Three-Level Inverter

In this section the extension of VSP^{2}TC to the three-level NPC voltage source inverter driving an IM (see Fig. 5.6) is presented. The implementation of VSP^{2}TC for the two-

level inverter—as presented in Section 5.4.2.1—is refined in order to meet all the control objectives, while exhibiting the same favorable performance during steady-state and transient operating conditions. Therefore, the regulation of the torque and the stator flux magnitude to their references, the balancing of the neutral point potential, i.e. the equal sharing of the dc-link voltage across the two capacitors, and the minimization of the torque ripple are to be achieved. To do so, an objective function that encompasses all the variables of concern is formulated and minimized in real-time. Based on the aforementioned optimization problem, a time instant that lies between the discrete time instants is calculated; at this instant the switches should change state.

The procedure is similar to the one presented in Section 5.4.2.1—though, appropriately modified—and it is repeated here briefly. Since the goal is to minimize the torque ripple, the variable switching point where the change of the switching state will take place, is to be calculated. This variable point is in the range $[0, 1]$, i.e. $n_{int} \in [0, 1]$, which means that the switchings occur at $kT_s \leq (k + n_{int}^{(k)})T_s \leq (k + 1)T_s$.

In a first step, the electromagnetic torque at step k is calculated according to (5.24b) from the measurements of the stator currents and the rotor speed. Following, and by applying the switching state calculated at the previous sampling interval $\boldsymbol{u}(k-1+n_{int}^{(k-1)})$, the evolution of the state, as given by (5.24a), is computed within the one-step horizon. Based on the predicted values of the state the prediction of the torque is derived, as well. Assuming that the slope m of the torque is constant within the sampling interval T_s, the predicted value of the torque, $T_e(k + 1)$, is related to the present one, $T_e(k)$, according to (5.30), and repeated here for the convenience of the reader

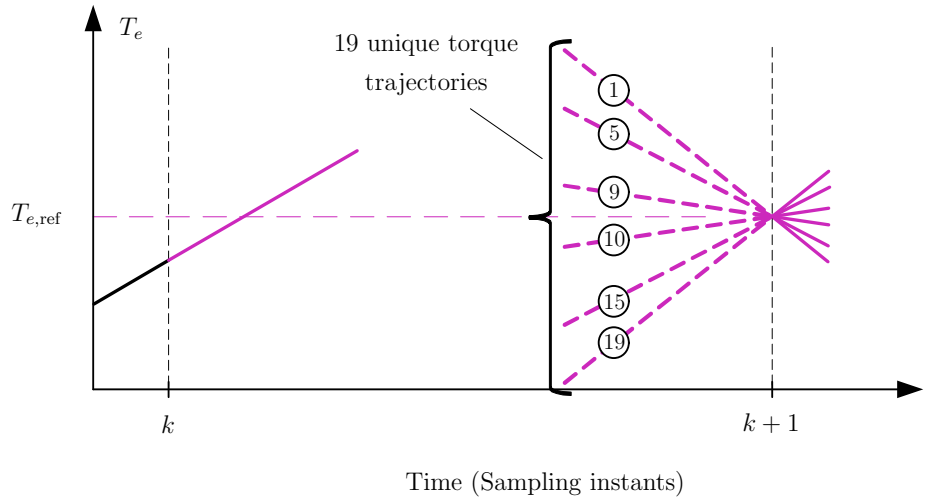
$$T_e(k + 1) = T_e(k) + m(k)T_s \,. \tag{5.35}$$

In a next step, the evolution of the state and of the output variables is calculated for all the possible voltage vectors, based on (5.24a). Thereby, the predicted value of the torque at step $k+1$ will differ, depending on the applied input vector $\boldsymbol{u}_z$, with $z \in \{0, 1, \ldots, 26\}$. Hence, (5.35) is modified in order to calculate the corresponding torque slopes

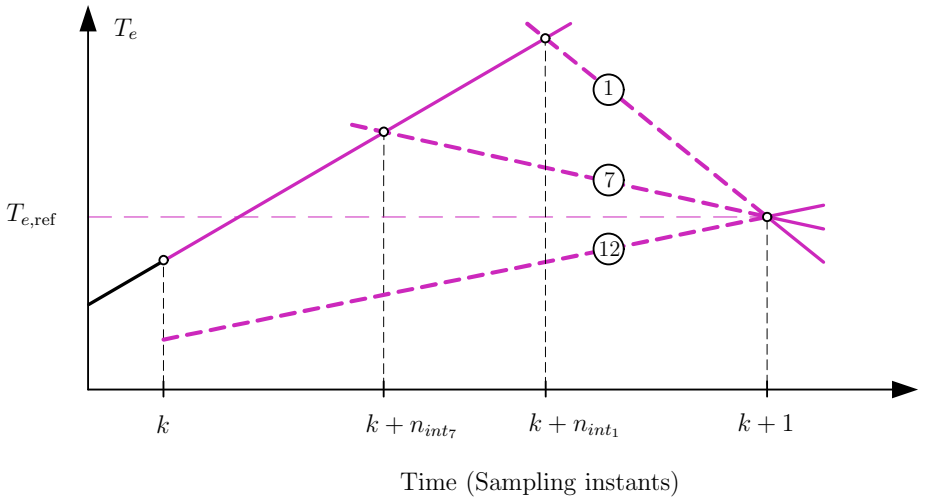
$$T_{e_z}(k + 1) = T_e(k) + m_z(k)T_s \,. \tag{5.36}$$

Exploiting the fact that the sampling interval is much smaller than the fundamental period $T_1 = 20\,\text{ms}$, i.e. $T_s \ll T_1$, the assumption that the torque slopes remain unchanged for the successive sampling interval is made, i.e. $m_z(k) = m_z(k + 1)$ (Fig. 5.14(a)).

In a subsequent step, the variable switching point $n_{int}^{(k)}$ is calculated, as explained in Section 5.4.2.1, i.e. by setting as goal at step $k + 1$ to get $T_{e_z}(k + 1) = T_{e,\text{ref}}$. Firstly, the intersection between the line segment formed by the actual $T_e(k)$ and the predicted torque $T_e(k + 1)$, and each one of the line segments resulting from the respective *possible* torques, $T_{e_z}(k + 1)$, is computed. Then, the corresponding potential switching points are



(a) The torque at k is calculated based on the applied switching state $\boldsymbol{u}(k) = \boldsymbol{u}(k-1+n_{int}^{(k-1)})$. 19 unique torque trajectories that correspond to the 27 different switching states are calculated at $k+1$.



(b) The point where the torques of steps k and $k+1$ intersect is calculated. Here, 3 out of 19 candidate torque trajectories are shown; torques (1) and (7) at $k+1$ intersect with $T_e(k)$, while torque (12) does not.

Figure 5.14: A variable switching point is calculated in order to minimize the torque slope. In (a) the calculation of the torque slopes and in (b) the calculation of the variable switching point are shown.

derived, since they are the projection of the computed intersections onto the time axis. Hence, the variable switching point is given by (5.32), and repeated here

$$t_z = \frac{T_{e,\mathrm{ref}} - T_e(k) - m_z(k+1)T_s}{m(k) - m_z(k+1)}\,. \tag{5.37}$$

Based on each one of the potential switching points, as calculated by (5.37), and the corresponding switching state, the prediction of the state and output variables is

recalculated using (5.24) for two successive steps, the intermediate step $k + n_{int_z}^{(k)}$, and step $k+1$. However, in (5.24) t_z is used instead of T_s as the prediction time interval for the first step, while for the second step the time interval is set equal to $T_s - t_z$.

In a last step the optimization problem is defined as

$$\begin{aligned} &\text{minimize} && J(k) \\ &\text{subject to} && (5.24)\,. \end{aligned} \tag{5.38}$$

In (5.38) the objective function is chosen to be

$$J(k) = \sum_{\xi \in \mathcal{S}} \left(||T_{e,\text{ref}} - T_e(k+\xi|k)||_2^2 + \lambda_1 ||\Psi_{s,\text{ref}} - \Psi_s(k+\xi|k)||_2^2 + \lambda_2 ||v_n(k+\xi|k)||_2^2 \right), \tag{5.39}$$

where the squared 2-norm is chosen in order to penalize deviations from the reference values more heavily. The set $\mathcal{S}$ is defined as $\mathcal{S} = \{n_{int}, 1\}$. In addition, the weighting factors $\lambda_1, \lambda_2 \in \mathbb{R}^+$ set the trade-off between the variables of interest, i.e. the electromagnetic torque error, the stator flux magnitude error and the neutral point potential. Note that λ_1 is tuned such that the same relative importance is assigned to both torque and flux control, while $\lambda_2 \ll \lambda_1$. Thus, λ_1 is set equal to the ratio between the nominal values of the electromagnetic torque $T_{e,\text{nom}}$, and the stator flux magnitude $\Psi_{s,\text{nom}}$ [31], i.e.

$$\lambda_1 = \left(\frac{T_{e,\text{nom}}}{\Psi_{s,\text{nom}}} \right)^2 . \tag{5.40}$$

The optimization problem (5.38) is solved in real-time within every T_s. The objective function (5.39) is evaluated for all possible switching states $\boldsymbol{u}(k)$; the switching state that results in the objective function with the minimum associated cost is the optimal $\boldsymbol{u}^*(k)$, and it is applied to the inverter at time instant $(k + n_{int}^{(k)})T_s$. Finally, at the next time-step, the whole procedure is repeated with new measurements or estimates. The overall block diagram consisting of the outer—speed—and the inner—constituted by VSP^{2}TC—loops is illustrated in Fig. 5.15.

5.5 Experimental Results

In this section experimental results of the proposed control strategy (VSP^{2}TC) are presented for both drives, i.e. the two-level inverter—IM drive, and the three-level NPC inverter—IM drive.

5.5.1 Drive with Two-Level Inverter

The proposed algorithm was tested in the laboratory. The experimental setup which consists of two 2.2 kW squirrel-cage IMs. One of the motors is used as load machine,

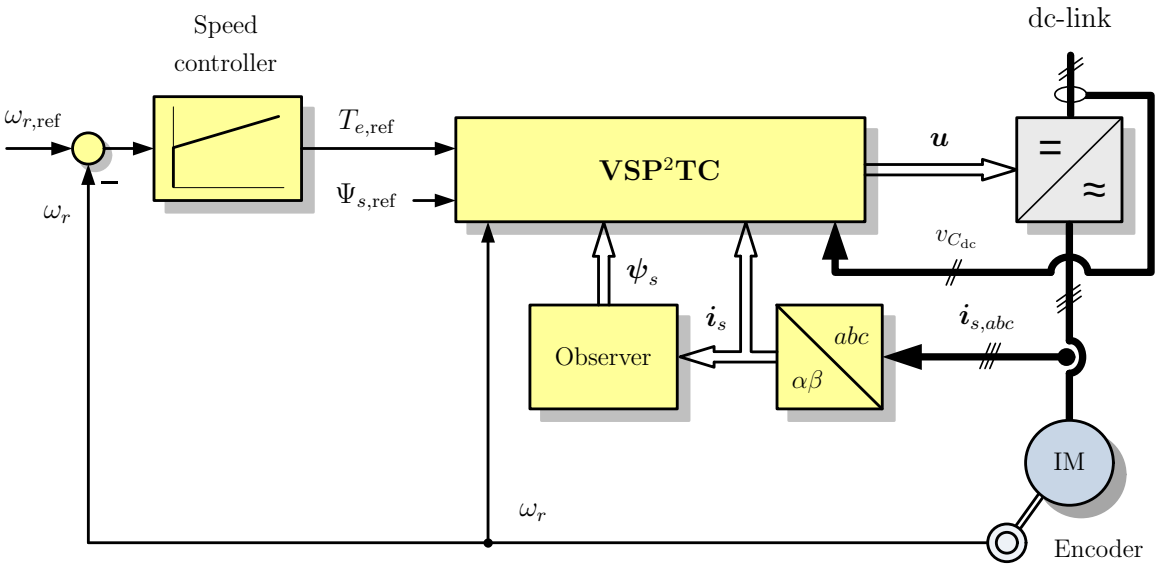


Figure 5.15: Block diagram of the variable switching point predictive torque controller (VSP²TC) for the three-level neutral point clamped (NPC) voltage source inverter driving an IM.

driven by a Danfoss VLT FC-302 3.0 kW inverter. The working machine is driven by a modified Seidel/Kollmorgen Servostar 600 14 kVA inverter which allows the user to give the gating signals directly via a suitable control system. The dc-links of both inverters are connected in order to avoid a frequent use of the break chopper resistor. The real-time computer with a 1.4 GHz Pentium CPU used for the experiments is described in [1]. The machine speed is measured via a 1024 points incremental encoder. Both algorithms are executed with a sampling interval $T_s = 61.44\,\mu$s. Table 5.1 shows the parameters of the experimental setup. The parameters of the working machine were measured with the Danfoss load inverter.

For PTC the objective function was chosen to be as the one presented in [102], i.e.

$$J(k) = \big(T_{e,\mathrm{ref}} - T_e(k+1)\big)^2 + \lambda\big(\Psi_{s,\mathrm{ref}} - \Psi_s(k+1)\big)^2 . \tag{5.41}$$

For the objective functions of both VSP²TC and PTC, given by (5.33) and (5.41), respectively, the weighting factor λ is tuned such that the same relative importance is assigned to both torque and flux control. Thereby, the weighting factor is chosen to be equal to the ratio between the nominal values of the electromagnetic torque $T_{e,\mathrm{nom}}$, and the stator flux magnitude $\Psi_{s,\mathrm{nom}}$ [31], i.e.

$$\lambda = \left(\frac{T_{e,\mathrm{nom}}}{\Psi_{s,\mathrm{nom}}}\right)^2 . \tag{5.42}$$

Finally, the same PI-speed controller has been used for both approaches in order to achieve a similar behavior with respect to the rotor speed.

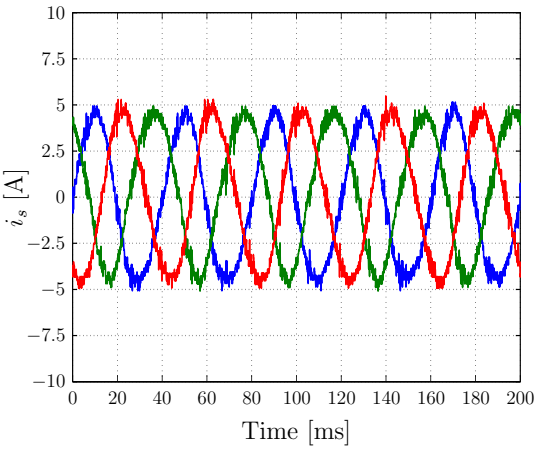
Table 5.1: Parameters of the experimental setup consisting of a two-level inverter and an induction motor.

Parameter		Value
Sampling interval	T_s	$61.44\,\mu$s
dc-link voltage	V_{dc}	582 V
Number of pole pairs	p	1
Nominal rotor speed	$\omega_{r,\mathrm{nom}}$	2772 rpm
Nominal torque	$T_{e,\mathrm{nom}}$	7.58 Nm
Nominal stator flux magnitude	$\Psi_{s,\mathrm{nom}}$	0.7 Wb
Stator resistance	r_s	$2.6827\,\Omega$
Rotor resistance	r_r	$2.1290\,\Omega$
Stator inductance	l_s	283.4 mH
Rotor inductance	l_r	283.4 mH
Mutual inductance	l_m	275.1 mH

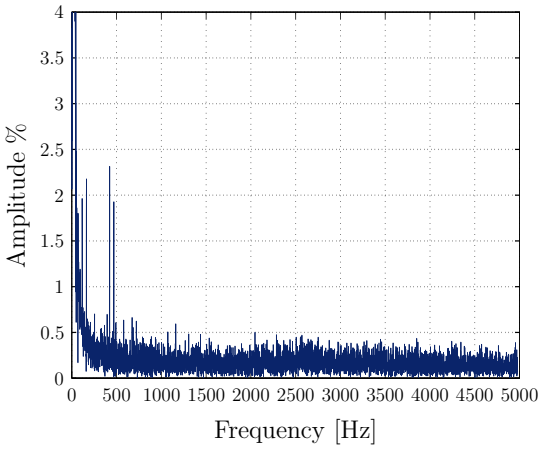
5.5.1.1 Steady-State Operation

The steady-state performance of the drive was examined for both VSP^{2}TC and PTC. The drive operates at half nominal speed, i.e. the fundamental frequency is $f_1 = 25\,\mathrm{Hz}$. The torque reference is set equal to $T_{e,\mathrm{ref}} = 4\,\mathrm{Nm}$, and the stator flux magnitude reference to $\Psi_{s,\mathrm{ref}} = 0.7\,\mathrm{Wb}$. The results are presented in Figs. 5.16 and 5.17 for VSP^{2}TC and PTC, respectively. As can be seen in Figs. 5.16(a) and 5.17(a), where the three phase stator currents are depicted, the VSP^{2}TC produces currents of lower total harmonic distortion (THD = 3.15%, Fig. 5.16(b)) than these which PTC delivers (THD = 4.11%, Fig. 5.17(b)). However, it should be mentioned that the switching frequency is not the same for both approaches, despite the fact that for both experiments the same sampling interval is used. For VSP^{2}TC the switching frequency is around $f_{sw} \approx 3.2\,\mathrm{kHz}$, while for PTC it is around $f_{sw} \approx 2.9\,\mathrm{kHz}$. That slight mismatch occurs because of the nature of the proposed strategy, as already explained in Section 5.4. The possibility for the inverter to select a different switching state within the interval leads to higher switching frequencies.

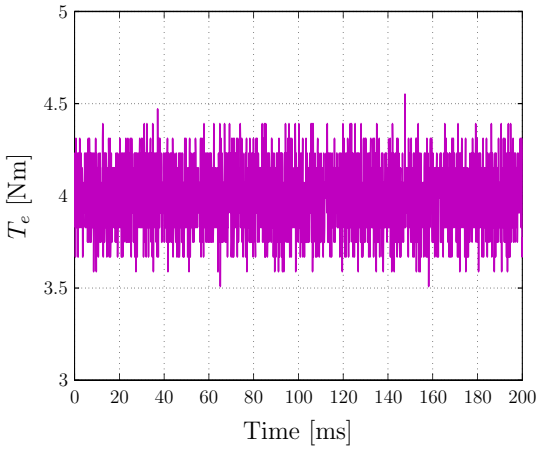
The results of the optimization over the torque ripple are presented in Fig. 5.16(c). As it is clearly shown, the proposed control strategy results in a significantly reduced torque ripple compared to PTC (Fig. 5.17(c)). Finally, the stator flux in the $\alpha\beta$ plane is depicted in Figs. 5.16(d) and 5.17(d); the flux stator magnitude is equal to its reference.



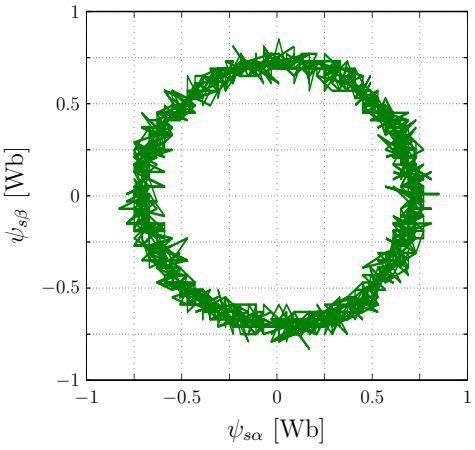
(a) Three-phase stator currents.



(b) Harmonic spectrum of the stator currents. The THD is 3.15%.



(c) Electromagnetic torque.

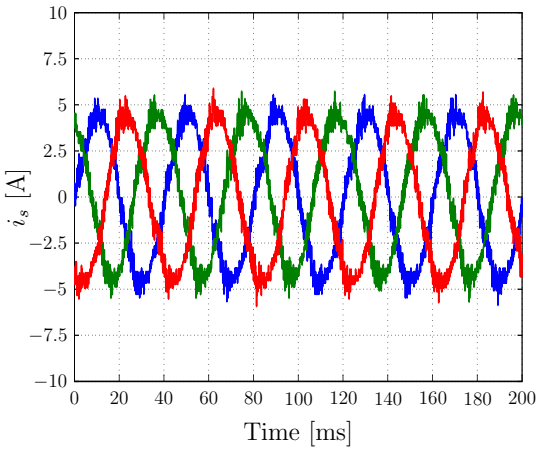


(d) Flux in $\alpha\beta$ plane.

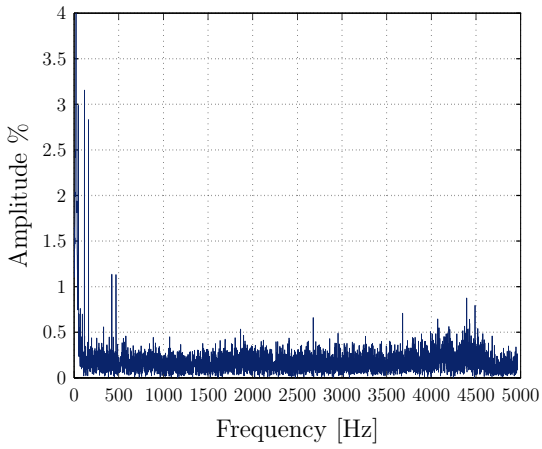
Figure 5.16: Experimental results of the proposed control strategy (VSP^{2}TC) for steady-state operation at half nominal speed ($f_1 = 25\,\text{Hz}$). The switching frequency is $f_{sw} \approx 3.2\,\text{kHz}$.

5.5.1.2 Torque Step Change Response

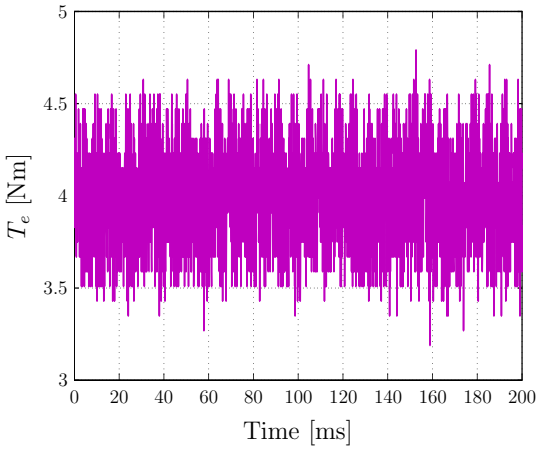
The performance of VSP^{2}TC during transients was also tested; the result is shown in Fig. 5.18. A step-up change in the torque reference takes place at $t \approx 3\,\text{ms}$ from $T_{e,\text{ref}} = 2\,\text{Nm}$ to $T_{e,\text{ref}} = 4\,\text{Nm}$. The controller reacts very fast to the reference torque change and rejects the disturbance in less than 0.5 ms (Fig. 5.18(a)). Furthermore, as can be seen in Fig. 5.18(b), the flux remains unaffected by the reference torque change. Finally, the torque response (Fig. 5.19(a)) for the same scenario was examined when the drive is controlled with PTC. Similar dynamic performance is observed, since the torque also reaches its new reference value very fast.



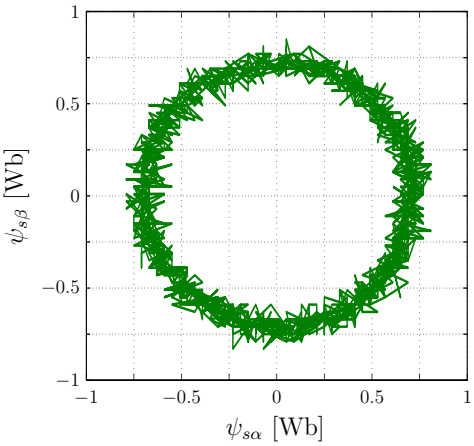
(a) Three-phase stator currents.



(b) Harmonic spectrum of the stator currents. The THD is 4.11%.



(c) Electromagnetic torque.



(d) Flux in $\alpha\beta$ plane.

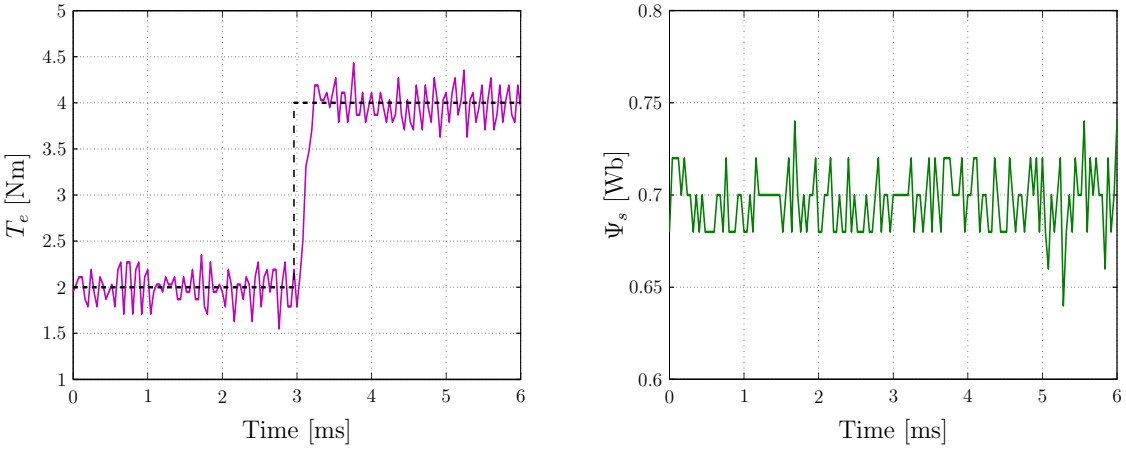
Figure 5.17: Experimental results of the predictive torque control (PTC) for steady-state operation at half nominal speed ($f_1 = 25\,\text{Hz}$). The switching frequency is $f_{sw} \approx 2.9\,\text{kHz}$.

5.5.1.3 Load Torque Impact

Following, a step change in the load torque was considered. While the machine is operated at $\omega_r = 2000\,\text{rpm}$, the load torque is stepped up from $T_\ell = 0\,\text{Nm}$ to $T_\ell = 4\,\text{Nm}$ at $t \approx 290\,\text{ms}$. As can be seen in Figs. 5.20 and 5.21, the dynamic behavior of the drive is similar for both algorithms, as expected, since the outer (speed) loop is the same. However, the ripple of the torque that VSP^{2}TC delivers (Fig. 5.20(a)) is clearly less than the torque ripple that PTC leads to (Fig. 5.21(a)).

5.5.1.4 Speed Reference Step Change

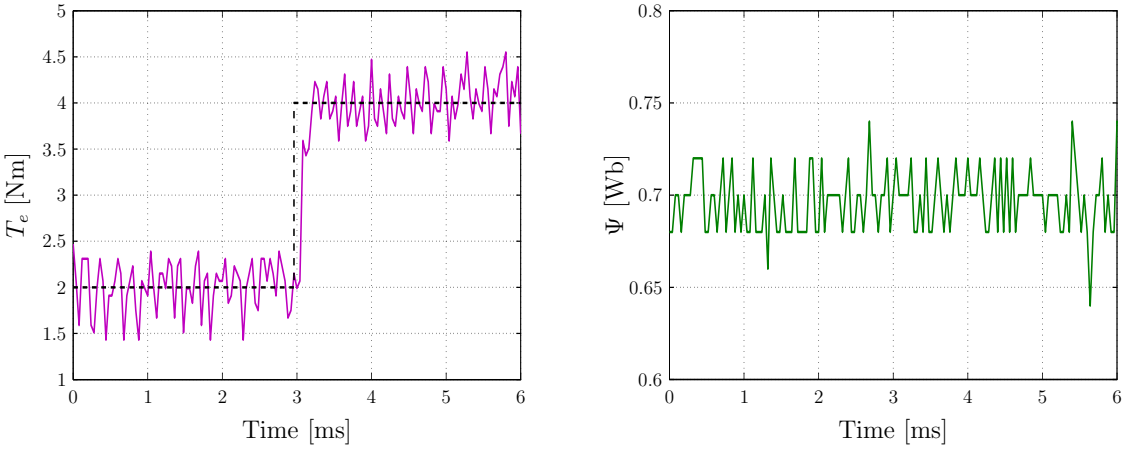
Another case to be analyzed is that of the transient behavior during a step-up change in the rotor speed reference from $\omega_{r,\text{ref}} = 1000\,\text{rpm}$ to $\omega_{r,\text{ref}} = 2000\,\text{rpm}$ at $t \approx 80\,\text{ms}$; for



(a) Electromagnetic torque (solid line) and its reference value (dashed line).

(b) Flux magnitude.

Figure 5.18: Experimental results with VSP^{2}TC for a step change in the electromagnetic torque reference at $t \approx 3\,\text{ms}$.

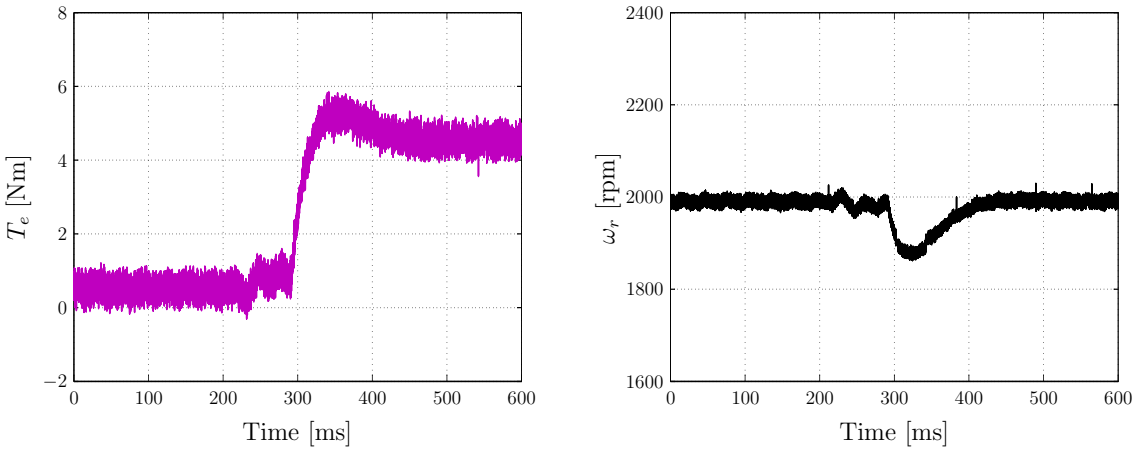


(a) Electromagnetic torque (solid line) and its reference value (dashed line).

(b) Flux magnitude.

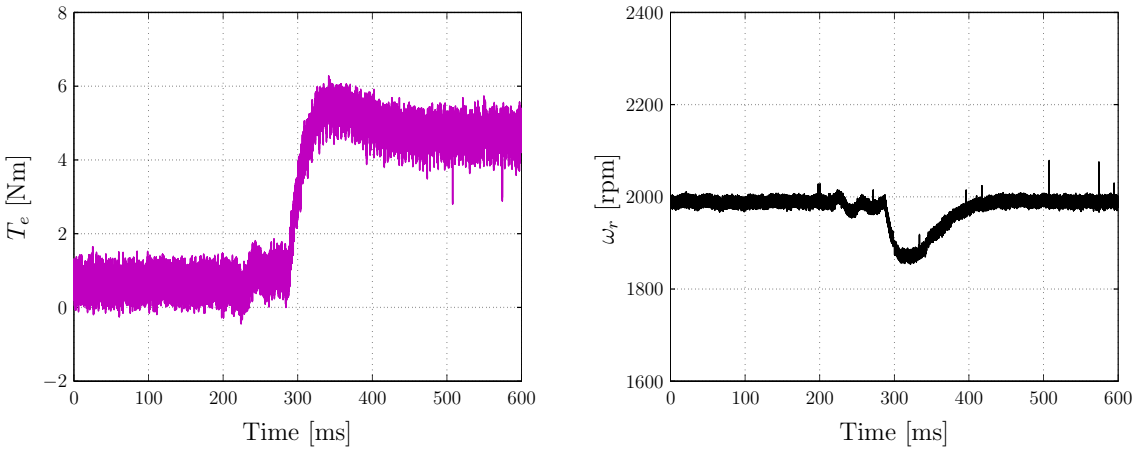
Figure 5.19: Experimental results with PTC for a step change in the electromagnetic torque reference at $t \approx 3\,\text{ms}$.

this case the IM operates at no load. For both VSP^{2}TC (Fig. 5.22) and PTC (Fig. 5.23) the rotor speed ω_r tracks its reference in about $t \approx 180\,\text{ms}$; the fast inner loop in both strategies is designed in the MPC framework, thus allowing an increased outer loop bandwidth without interference between the two loops [120]. Moreover, the additional control objective of VSP^{2}TC, i.e. the torque ripple minimization, is met, see Fig. 5.22(a) for VSP^{2}TC compared to Fig. 5.23(a) for PTC.



(a) Electromagnetic torque. (b) Rotor speed.

Figure 5.20: Experimental results with VSP^{2}TC for a step change in the load torque at $t \approx 290\,\text{ms}$.

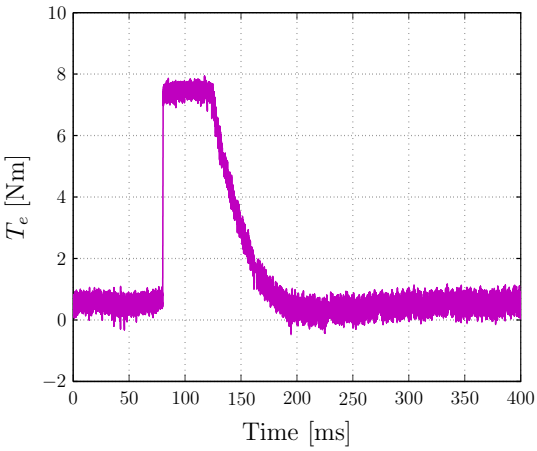


(a) Electromagnetic torque. (b) Rotor speed.

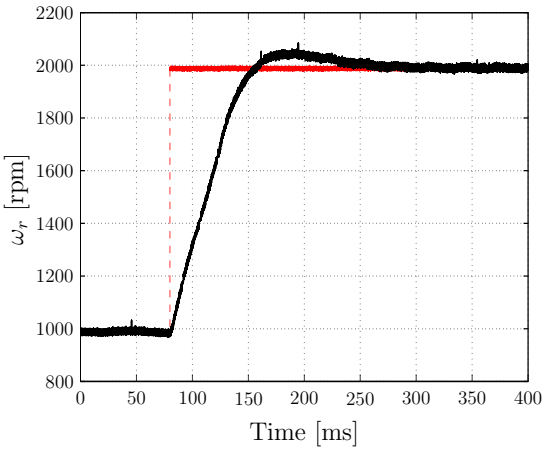
Figure 5.21: Experimental results with PTC for a step change in the load torque at $t \approx 290\,\text{ms}$.

5.5.1.5 Speed Reversal

The last case examined is a speed reversal maneuver occurring at $t \approx 280\,\text{ms}$, from positive nominal $\omega_{r,\text{nom}}$ to negative nominal speed $-\omega_{r,\text{nom}}$, under no-load operation. As can be seen in Figs. 5.24 and 5.25, the additional control objective of VSP^{2}TC does not deteriorate the dynamic performance of the drive. On the contrary, for both algorithms the rotor speed reaches its final value $-\omega_{r,\text{nom}}$ in about $t \approx 300\,\text{ms}$ (Figs. 5.24(b) and 5.25(b)), while the ripple of the torque that VSP^{2}TC produces is significantly reduced compared to that of PTC (Figs. 5.24(a) and 5.25(a), respectively).

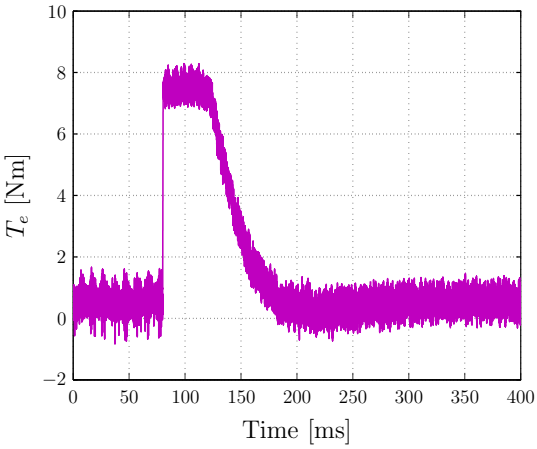


(a) Electromagnetic torque.

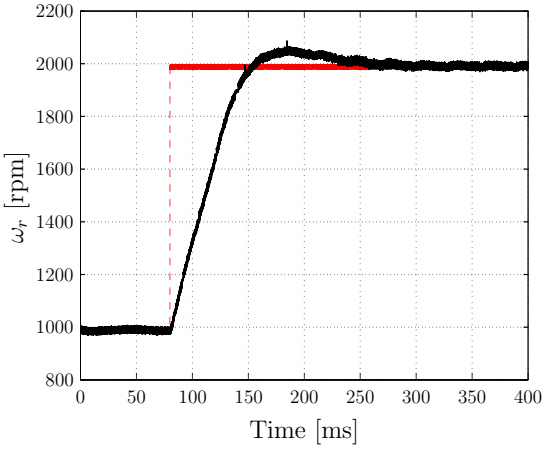


(b) Rotor speed (solid line) and its reference value (dashed line).

Figure 5.22: Experimental results with VSP^{2}TC for a step change in the rotor speed reference at $t \approx 80\,\text{ms}$.



(a) Electromagnetic torque.



(b) Rotor speed (solid line) and its reference value (dashed line).

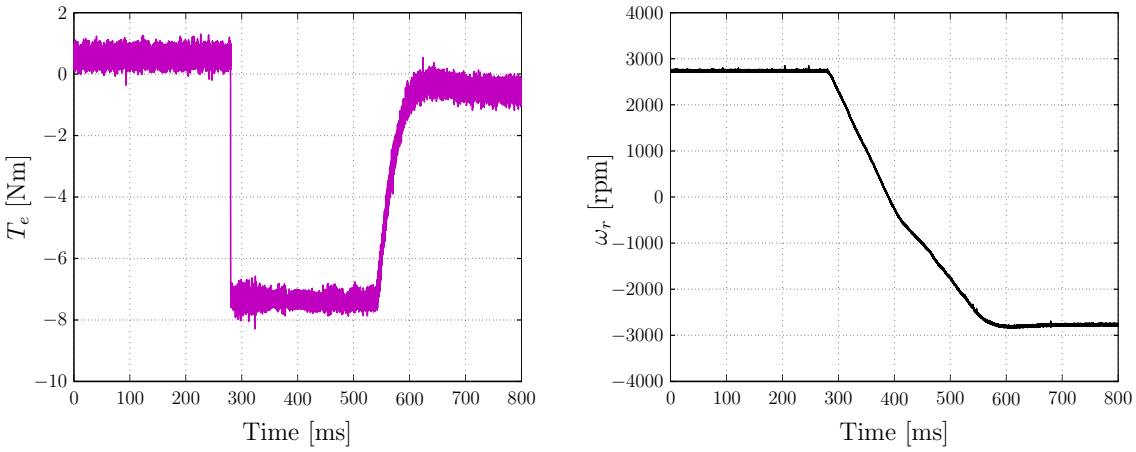
Figure 5.23: Experimental results with PTC for a step change in the rotor speed reference at $t \approx 80\,\text{ms}$.

5.5.2 Drive with Three-Level Inverter

The performance of a three-level NPC inverter driving a 2.2 kW squirrel-cage IM under no load was tested with PTC and VSP^{2}TC. The parameters of the experimental setup are listed in Table 5.2. For the experiment a real-time computer system was used with a 3.5 GHz Pentium 4 processor. Both algorithms were executed with a sampling interval $T_s = 200\,\mu\text{s}$.

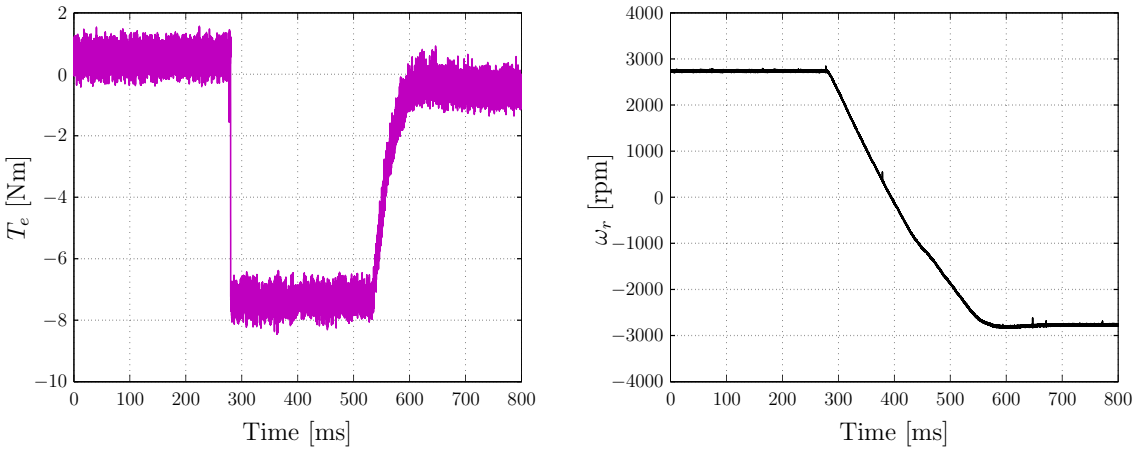
For PTC the objective function was chosen to be

$$J(k) = \big(T_{e,\text{ref}} - T_e(k+1)\big)^2 + \lambda_1\big(\Psi_{s,\text{ref}} - \Psi_s(k+1)\big)^2 + \lambda_2\big(v_n(k+1|k)\big)^2 . \qquad (5.43)$$



(a) Electromagnetic torque. (b) Rotor speed.

Figure 5.24: Experimental results with VSP^{2}TC for a speed reversal maneuver at $t \approx 280\,\text{ms}$.



(a) Electromagnetic torque. (b) Rotor speed.

Figure 5.25: Experimental results with PTC for a speed reversal maneuver at $t \approx 280\,\text{ms}$.

As mentioned in Section 5.4.2.2, λ_1 in the formulated objective functions of both VSP^{2}TC and PTC, given by (5.39) and (5.43), respectively, is tuned such that the same relative importance is assigned to both torque and flux control (see (5.40)), and λ_2 is chosen to be $\lambda_2 \ll \lambda_1$. Moreover, the same PI-speed controller has been used for both approaches in order to achieve a similar behavior with respect to the rotor speed.

5.5.2.1 Steady-State Operation

Initially, the torque reference is set equal to $T_{e,\text{ref}} \approx 0\,\text{Nm}$, and the stator flux magnitude reference to $\Psi_{s,\text{ref}} = 0.78\,\text{Wb}$. The torque and stator flux (in the $\alpha\beta$ plane) waveforms recorded in the experimental setup with the drive system controlled by VSP^{2}TC and PTC are shown in Figs. 5.26 and 5.27, respectively. As can be seen, the VSP^{2}TC significantly

Table 5.2: Parameters of the experimental setup consisting of a three-level neutral point clamped inverter and an induction motor.

Parameter		Value
Sampling interval	T_s	$200\,\mu$s
dc-link voltage	V_{dc}	550 V
Number of pole pairs	p	1
Nominal rotor speed	$\omega_{r,\mathrm{nom}}$	2772 rpm
Nominal torque	$T_{e,\mathrm{nom}}$	7 Nm
Nominal stator flux magnitude	$\Psi_{s,\mathrm{nom}}$	0.78 Wb
Stator resistance	r_s	$2.1294\,\Omega$
Rotor resistance	r_r	$2.2773\,\Omega$
Stator inductance	l_s	350.47 mH
Rotor inductance	l_r	350.47 mH
Mutual inductance	l_m	340.42 mH

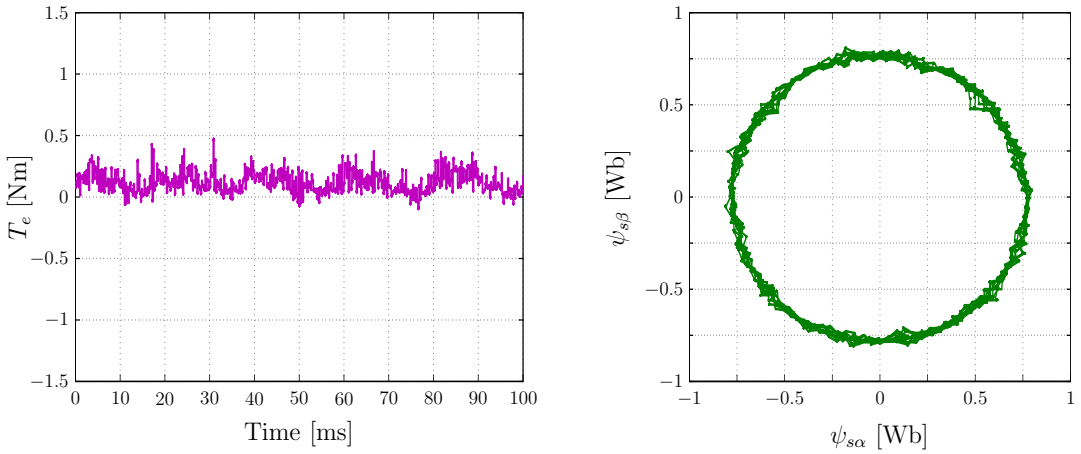
reduces the torque ripple (Fig. 5.26(a)), compared to PTC (Fig. 5.27(a)). It should be pointed out, though, that—as expected from the analysis presented above—the switching frequency is not the same for both approaches, despite the fact that for both experiments the same T_s is used. The switching frequency for VSP^{2}TC is around $f_{sw} \approx 950$ Hz, while for PTC it is around $f_{sw} \approx 750$ Hz.

5.5.2.2 Torque Step Change Response

At time $t \approx 20$ ms the torque reference is stepped up from $T_{e,\mathrm{ref}} \approx 0$ Nm to $T_{e,\mathrm{ref}} \approx 7$ Nm, while the stator flux reference is kept equal to $\Psi_{s,\mathrm{ref}} = 0.78$ Wb. The dynamic performance of VSP^{2}TC is shown in Fig. 5.28; the one of PTC in Fig. 5.29. The torque ripple of VSP^{2}TC (Fig. 5.28(a)) is significantly reduced compared to that of PTC (Fig. 5.29(a)). Furthermore, the transient torque response is not deteriorated by the use of the variable switching point. Finally, the stator flux magnitude (see Figs. 5.28(b) and 5.29(b)) is also not influenced for both controllers.

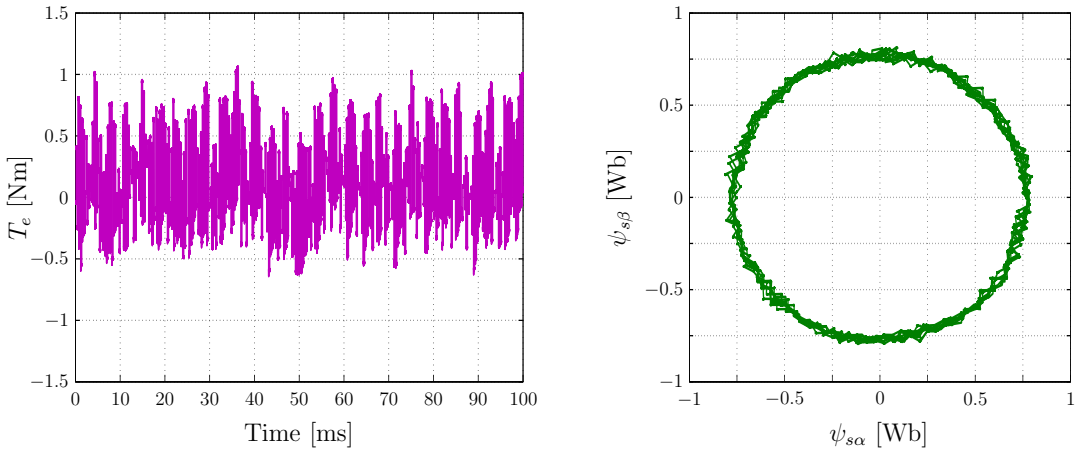
5.5.2.3 Speed Reference Step Change

Following, a step change in the rotor speed reference was considered. The rotor speed reference is stepped up from $\omega_{r,\mathrm{ref}} \approx 1300$ rpm to $\omega_{r,\mathrm{ref}} \approx 2700$ rpm at $t \approx 120$ ms. Both approaches (VSP^{2}TC and PTC shown in Figs. 5.30 and 5.31, respectively) yield a similar



(a) Electromagnetic torque. (b) Flux in $\alpha\beta$ plane.

Figure 5.26: Experimental results of the variable switching point predictive torque control (VSP^{2}TC) for steady-state operation under no load. The switching frequency is $f_{sw} \approx 950\,\text{Hz}$.



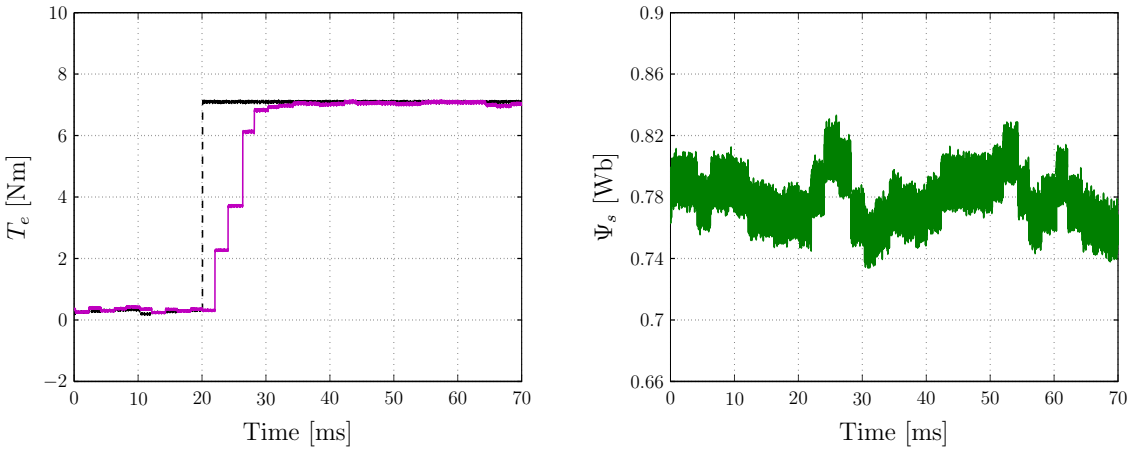
(a) Electromagnetic torque. (b) Flux in $\alpha\beta$ plane.

Figure 5.27: Experimental results of the predictive torque control (PTC) for steady-state operation under no load. The switching frequency is $f_{sw} \approx 750\,\text{Hz}$.

behavior during this transient since the rotor speed ω_r tracks its reference in about $t \approx 150\,\text{ms}$. Nonetheless, the torque ripple that VSP^{2}TC produces (Fig. 5.30(a)) is less compared to that of PTC (Fig. 5.31(a)).

5.5.2.4 Speed Reversal

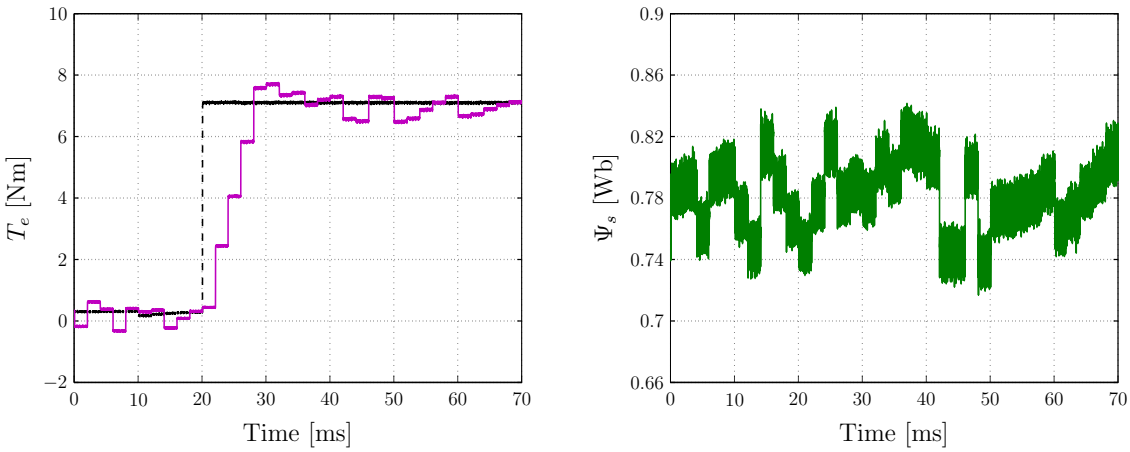
The last case analyzed is that of a speed reversal maneuver from positive nominal $\omega_{r,\text{nom}}$ to negative nominal speed $-\omega_{r,\text{nom}}$ occurring at $t \approx 350\,\text{ms}$. The additional control objective of VSP^{2}TC, i.e. the minimization of the torque ripple, is met, see Fig. 5.32(a). However, it does not deteriorate the dynamic performance of the drive since for both controllers the rotor speed reaches its final value in about $t \approx 200\,\text{ms}$ (Figs. 5.32(b) and 5.33(b)).



(a) Electromagnetic torque (solid magenta line) and its reference value (dashed black line).

(b) Flux magnitude.

Figure 5.28: Experimental results with VSP^{2}TC for a step change in the electromagnetic torque reference.



(a) Electromagnetic torque (solid magenta line) and its reference value (dashed black line).

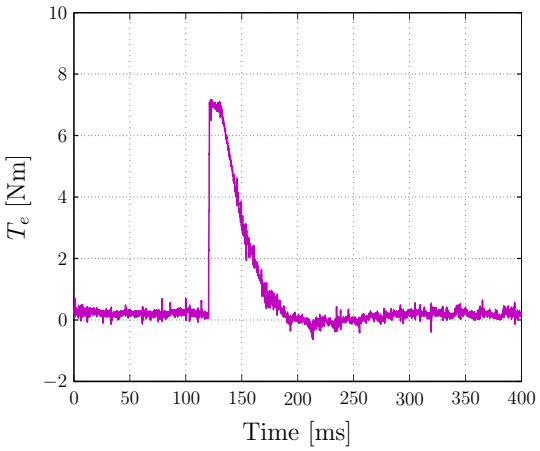
(b) Flux magnitude.

Figure 5.29: Experimental results with PTC for a step change in the electromagnetic torque reference.

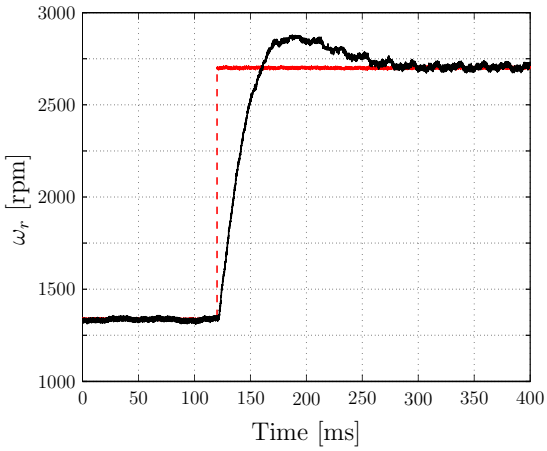
Finally, in Figs. 5.32(c) and 5.33(c) the dc-link voltage is shown.

5.6 Conclusions

In this chapter, a control algorithm, namely the variable switching point predictive torque control (VSP^{2}TC), was introduced and applied to two different drive systems. The first system is consisting of a two-level inverter and an induction machine (IM), and the second of a three-level neutral point clamped (NPC) inverter and an IM. By selecting a different switching state within the sampling interval the goal is to reduce the torque ripple, while

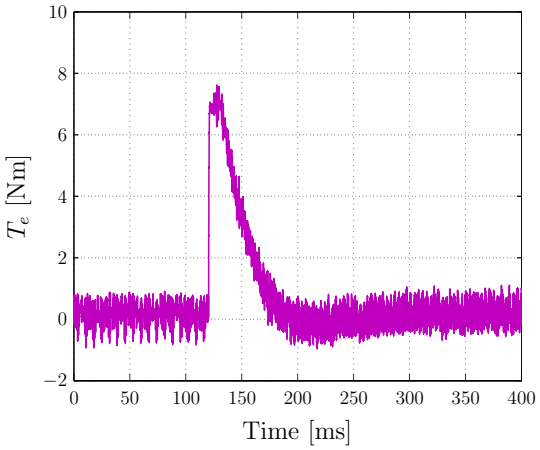


(a) Electromagnetic torque.

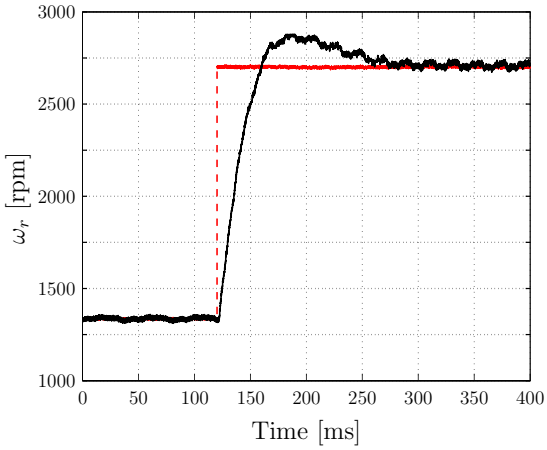


(b) Rotor speed (solid black line) and its reference value (dashed red line).

Figure 5.30: Experimental results with VSP^{2}TC for a step change in the rotor speed reference.



(a) Electromagnetic torque.

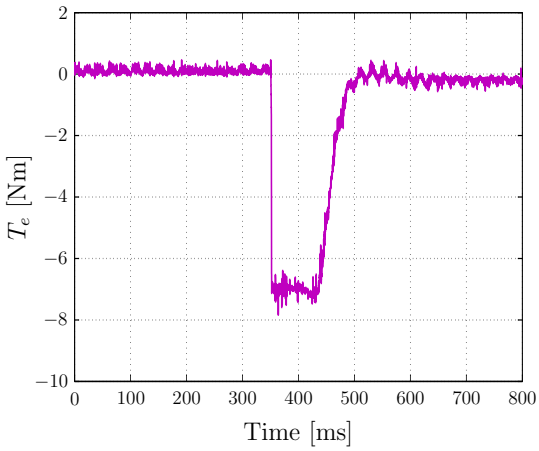


(b) Rotor speed (solid black line) and its reference value (dashed red line).

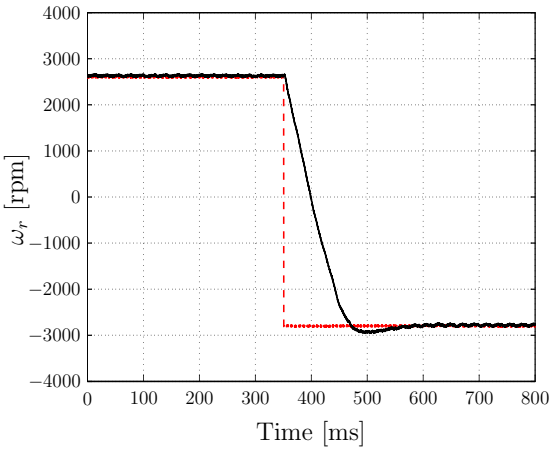
Figure 5.31: Experimental results with PTC for a step change in the rotor speed reference.

achieving zero steady-state torque and stator flux tracking errors, as well as—for the case of the three-level NPC inverter—neutral point potential balancing.

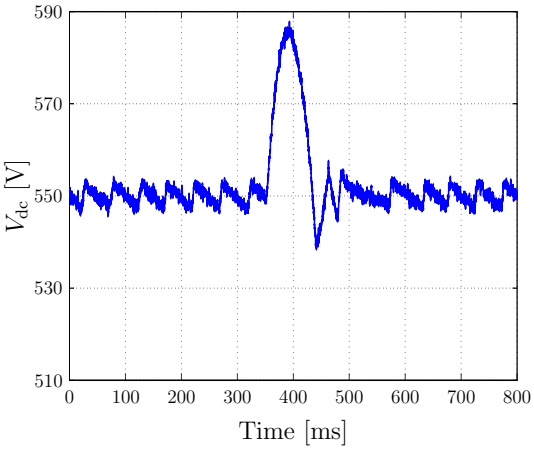
In order to meet the control objectives the variable switching point is calculated according to an optimization problem formulated to minimize the torque ripple. The advantages of the proposed method, among others, include the design simplicity, the straightforward implementation procedure, the fast dynamics that MPC can provide, and the inherent robustness. Furthermore, since the prediction horizon is kept as short as possible, i.e. $N = 1$, the computational complexity of the proposed approach is limited, making its implementation in a drive system possible. Finally, thanks to its flexibility it can be easily extended to different types of machines and be adapted for other control



(a) Electromagnetic torque.



(b) Rotor speed (solid black line) and its reference value (dashed red line).

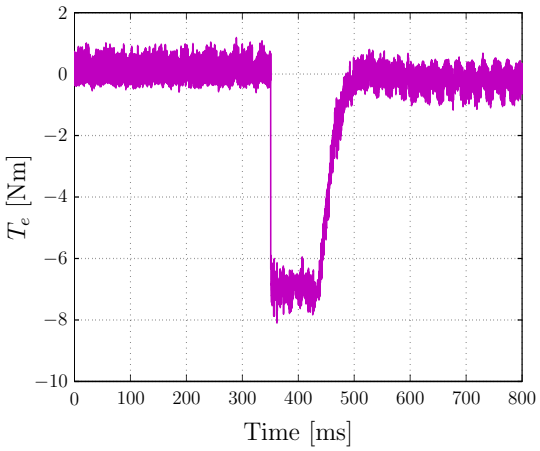


(c) Dc-link voltage.

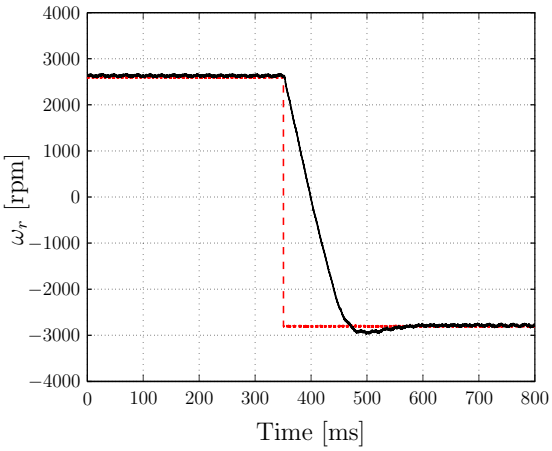
Figure 5.32: Experimental results with VSP^{2}TC for a speed reversal maneuver.

tasks.

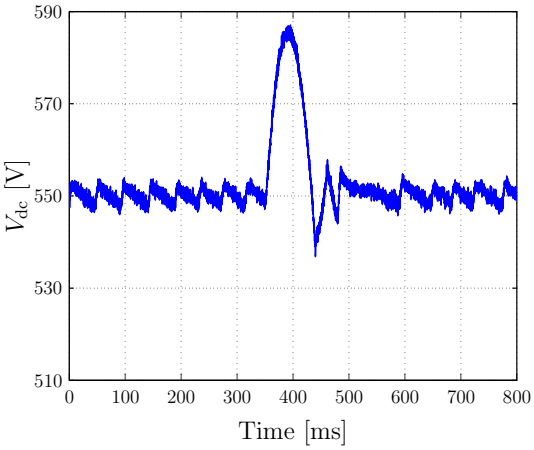
Nevertheless, the VSP^{2}TC method comes with a slightly increased switching frequency compared to predictive torque control (PTC), as proposed in [29] and [102]. However, this is not a major disadvantage; since the application target of the introduced strategy is the low voltage (LV) drives field, the switching losses are not so important compared to medium- (MV) or high voltage (HV) drives where they are of paramount importance. Moreover, it should be pointed out, that this higher switching frequency can be achieved with the same sampling interval T_s for both algorithms (VSP^{2}TC and PTC), implying that there is no need for more costly hardware. Therefore, the proposed method can be successfully applied to drive systems up to a few kW, and to significantly improve the control result. This is verified by the presented experimental results; the favorable performance of VSP^{2}TC under not only steady-state, but also transient operating conditions



(a) Electromagnetic torque.



(b) Rotor speed (solid black line) and its reference value (dashed red line).



(c) Dc-link voltage.

Figure 5.33: Experimental results with PTC for a speed reversal maneuver.

for both drive systems examined, is clearly demonstrated.

Future Research

A very interesting topic is to extend the prediction horizon, but at the same time, with keeping the computational complexity modest. In this way, the control result will be further improved, while the implementation of the algorithm in a real-time system will still be possible. Furthermore, tackling the control problem as a current ripple minimization problem, instead of a torque one, is very intriguing.

Part III

Appendix

Appendix A

Nomenclature

Mathematical Definitions

General

a	scalar
$\boldsymbol{a}$	vector
$\boldsymbol{A}$	matrix
$\cdot$	general placeholder (for any variable)
$\{\cdot, \ldots\}$	set
$[\cdot, \ldots]$	matrix (or row vector)
$=$	equality
$\neq$	inequality, is not equal to
$<$ ($\leq$)	inequality, is less than (or equal to)
$>$ ($\geq$)	inequality, is greater than (or equal to)
$\ldots$	horizontal ellipsis, "and so forth"
$\vdots$	vertical ellipsis, "and so forth"
$\ddots$	diagonal ellipsis, "and so forth"
$:=$	definition
$\vert$	such that
$:$	such that
$\in$	is element of, belongs to
$\forall$	for all
$\rightarrow$	mapping

Sets

$\mathbb{N}$	natural numbers
$\mathbb{N}^+$	natural numbers without zero
$\mathbb{Z}$	integers
$\mathbb{R}$	real numbers
$\mathbb{R}^+$	nonnegative real numbers
$\mathbb{R}^n$	set of real vectors with n elements
$\mathbb{R}^{n\times m}$	set of real matrices with n rows and m columns

Operation with Logic Variables

$\Rightarrow$	implies
$\wedge$	and
$\vee$	or

Operation with Vectors

$\mathbb{0}$	vector of zeros, $\mathbb{0} = [0\ 0 \ldots 0]^T \in \mathbb{R}^n$
$\boldsymbol{a}^T$	row vector
$\preceq$	componentwise inequality ($u \preceq v$ means $u_i \leq v_i$ for $i = 1, \ldots, m$), similar for $\prec, \succ, \succeq$
$\lvert\boldsymbol{a}\rvert$	componentwise absolute value
$\lvert\lvert\boldsymbol{a}\rvert\rvert_\ell$	ℓ-norm ($\ell \geq 1$), $\lvert\lvert x\rvert\rvert_\ell = (\lvert x_1\rvert^\ell + \ldots + \lvert x_n\rvert^\ell)^{1/\ell}$

Operation with Matrices

$\boldsymbol{I}$	identity matrix (of appropriate dimension)
$\mathbf{0}$	zero matrix (of appropriate dimension)
$\mathbf{S}^n$	set of symmetric $n \times n$ matrices ($\mathbf{S}^n = \{\boldsymbol{X} \in \mathbb{R}^{n\times n} \mid \boldsymbol{X} = \boldsymbol{X}^T\}$)
$\mathbf{S}^n_+$	set of symmetric positive semidefinite matrices ($\mathbf{S}^n_+ = \{\boldsymbol{X} \in \mathbf{S}^n \mid \boldsymbol{X} \succeq 0\}$)
$\boldsymbol{A}^T$	matrix transpose
$\boldsymbol{A}^{-1}$	inverse of a square matrix

Operation with Sets

$\emptyset$	empty set
$\cap$	intersection
$\cup$	union

$(\subset)$ $\subseteq$	(strict) subset
$(\supset)$ $\supseteq$	(strict) superset

Optimization

min	minimum
max	maximum
inf	infimum
sup	supremum

Variables

$\boldsymbol{A}$	continuous-time state-space matrix, in ac drives
$\boldsymbol{A}_1, \boldsymbol{A}_2$	continuous-time state-space matrices, in dc-dc converter
$\boldsymbol{A}_1, \boldsymbol{A}_2$	continuous-time state-space matrices, in CHB multilevel rectifier
$\boldsymbol{A}_d$	discrete-time state-space matrix, in CHB multilevel rectifier
$\boldsymbol{B}$	continuous-time state-space matrix, in dc-dc converter
$\boldsymbol{B}$	continuous-time state-space matrix, in CHB multilevel rectifier
$\boldsymbol{B}, \boldsymbol{B}_1, \boldsymbol{B}_2$	continuous-time state-space matrices, in ac drives
$\boldsymbol{B}_d$	discrete-time state-space matrix, in CHB multilevel rectifier
$\boldsymbol{C}$	continuous-time state-space matrix, in dc-dc converter
$\boldsymbol{C}$	continuous-time state-space matrix, in CHB multilevel rectifier
$\boldsymbol{C}$	continuous-time state-space matrix, in ac drives
$\boldsymbol{C}_d$	discrete-time state-space matrix, in CHB multilevel rectifier
$C_{\boldsymbol{dc}}$	dc-link capacitor, in ac drives
C_o	capacitor, in dc-dc converter
C_{o_i}	capacitor of the ith cell, in CHB multilevel rectifier
d	duty cycle
$d_{\boldsymbol{aux}}$	auxiliary variable, in dc-dc converter
$\boldsymbol{D}$	continuous-time state-space matrix, in ac drives
$\boldsymbol{E}_1, \ldots, \boldsymbol{E}_4$	discrete-time state-space matrices, in dc-dc converter
$\boldsymbol{E}_{1a}, \ldots, \boldsymbol{E}_{4a}$	stochastic discrete-time state-space matrices, in dc-dc converter
f_{sw}	switching frequency
$\boldsymbol{F}_1, \boldsymbol{F}_2, \boldsymbol{F}_3$	discrete-time state-space matrices, in dc-dc converter
$\boldsymbol{F}_{1a}, \ldots, \boldsymbol{F}_{4a}$	stochastic discrete-time state-space matrices, in dc-dc converter
$\boldsymbol{G}$	discrete-time state-space matrix, in dc-dc converter
$\boldsymbol{G}_a$	stochastic discrete-time state-space matrix, in dc-dc converter
$i_e, \hat{i}_e$	inductor current disturbance, and estimated state, in dc-dc converter
i_L	inductor current, in dc-dc converter
i_s	input current, in CHB multilevel rectifier
$\boldsymbol{i}_{s\alpha\beta}$	stator current in $\alpha\beta$ plane, in ac drives
i_{sx}	stator current of phase x, with $x \in \{a, b, c\}$, in ac drives
i_{o_i}	output current of the ith cell, in CHB multilevel rectifier
J	objective function
H	inertia, in ac drives
k	discrete-time instants, $k \in \mathbb{N}^+$
$k_{\boldsymbol{pr}_i}, k_{\boldsymbol{int}_i}$	proportional, and integral gain of the ith PI-controller, in CHB multilevel

	rectifier
$\boldsymbol{K}$	transformation matrix, in ac drives
$K_1, \ldots, K_4$	Kalman gains, in dc-dc converter
ℓ	time-step within prediction horizon
l_s, l_r, l_m	stator, rotor, and mutual inductances, in ac drives
L	inductor, in dc-dc converter
L	inductor, in CHB multilevel rectifier
m	torque slope, in ac drives
M	number of samples in one period of the second harmonic (relative to the input voltage frequency) of the output voltage, in CHB multilevel rectifier
n	number of modules (cells), in CHB multilevel rectifier
n_{int}	time instant within the sampling interval, in ac drives
n_s	sampling interval multiple of T_s in move blocking scheme, in dc-dc converter
N	length of prediction horizon
N_1	number of steps in the first part of the horizon, which are sampled with T_s in move blocking scheme, in dc-dc converter
N_2	number of steps in the last part of the horizon, which are sampled with a multiple of T_s in move blocking scheme, in dc-dc converter
p	pole pair number, in ac drives
$\boldsymbol{Q}$	noise covariance matrix, in dc-dc converter
r_s, r_r	stator, and rotor resistances, in ac drives
$\boldsymbol{R}$	noise covariance matrix, in dc-dc converter
R, R_L	load, and inductor internal resistors, in dc-dc converter
R_L	inductor internal resistor, in CHB multilevel rectifier
t_z	projection of the intersection point onto the x-axis, in ac drives
T_1	fundamental period, in ac drives
T_e	electromagnetic torque, in ac drives
T_{i1}, T_{i2}	pair of switches of the ith cell, in CHB multilevel rectifier
T_ℓ	mechanical load torque, in ac drives
T_s	sampling interval
$\boldsymbol{u}$	input vector
$\boldsymbol{u}_{abc}$	vector of switching states of inverter in abc plane, in ac drives
$\boldsymbol{U}$	sequence of input vectors within the prediction horizon
v_{ab}	reflected multilevel voltage to the ac side, in CHB multilevel rectifier
$\boldsymbol{v}_{\alpha\beta}$	voltage vector in $\alpha\beta$ plane, in ac drives

$v_e, \hat{v}_o$	output voltage disturbance, and estimated state, in dc-dc converter
v_n	neutral point potential, in ac drives
$v_{\boldsymbol{ph},x}$	voltage of phase x, with $x \in \{a, b, c\}$, in ac drives
v_s, v_o	input, and output voltages, in dc-dc converter
v_s, v_{o_i}	input, and output voltages of the ith cell, in CHB multilevel rectifier
$\bar{v}_{o_i}$	dc component of the output voltage of the ith cell, in CHB multilevel rectifier
$V_{\boldsymbol{dc}}$	dc-link voltage, in ac drives
$\boldsymbol{w}$	disturbance vector, in CHB multilevel rectifier
$\boldsymbol{x}$	state vector
$\boldsymbol{x}_a, \hat{\boldsymbol{x}}_a$	augmented, and estimated state vector, in dc-dc converter
$\boldsymbol{y}$	output vector
$\boldsymbol{\Gamma}_1, \boldsymbol{\Gamma}_2, \boldsymbol{\Gamma}_3$	continuous-time state-space matrices, in dc-dc converter
δ	rotor flux angle, in ac drives
$\boldsymbol{\Delta}$	continuous-time state-space matrix, in dc-dc converter
λ	weighting factor
$\boldsymbol{\nu}$	measurement noise, in dc-dc converter
$\boldsymbol{\xi}$	process noise, in dc-dc converter
τ_1, τ_2	time-instants within the sampling interval, in dc-dc converter
ϕ	angle between a-axis of the three-phase abc system and d-axis of the $dq0$ reference frame, and angular position of rotating reference frame, in ac drives
$\boldsymbol{\psi}_r, \boldsymbol{\psi}_s$	stator, and rotor flux in dq (or $\alpha\beta$) plane, in ac drives
Ψ_s	magnitude of stator flux, in ac drives
ω_{fr}	angular speed of frame, in ac drives
ω_r	rotor rotational speed, in ac drives

Acronyms

CCM	Continuous Conduction Mode
CHB	Cascaded H-Bridge
DCM	Discontinuous Conduction Mode
DTC	Direct Torque Control
FOC	Field Oriented Control
FC	Flying Capacitor
IM	Induction Machine/Motor
LP	Linear Program(ming)
LV	Low Voltage
MILP	Mixed-Integer Linear Program(ming)
MIMO	Multiple-Input Multiple-Output
MIP	Mixed-Integer Program(ming)
MIQP	Mixed-Integer Quadratic Program(ming)
MLD	Mixed Logical Dynamical
MPC	Model Predictive Control
NPC	Neutral Point Clamped
p.u.	per unit
PID	Proportional-Integral-Derivative
PLL	Phase-Locked Loop
PTC	Predictive Torque Control
PWM	Pulse Width Modulation
QP	Quadratic Program(ming)
SISO	Single-Input Single-Output
THD	Total Harmonic Distortion
VSP^{2}TC	Variable Switching Point Predictive Torque Control

Bibliography

[1] N. Al-Sheakh Ameen, A. A. Naassani, and R. M. Kennel. Design of a digital system dedicated for electrical drive applications. *EPE J.*, 20(4):37–44, Dec. 2010.

[2] R. Alur, C. Courcoubetis, T. A. Henzinger, and P. H. Ho. Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems. In R. Grossman, A. Nerode, A. Ravn, and H. Rischel, editors, *Hybrid Syst.*, volume 736 of *LNCS*, pages 209–229. Springer-Verlag, 1993.

[3] J. Álvares-Ramírez, I. Cervantes, G. Espinosa-Pérez, P. Maya, and A. Morales. A stable design of PI control for dc-dc converters with an RHS zero. *IEEE Trans. Circuits Syst. I*, 48(1):103–106, Jan. 2001.

[4] A. D. Ames, A. Abate, and S. Sastry. Sufficient conditions for the existence of Zeno behavior in a class of nonlinear hybrid systems via constant approximations. In *Proc. IEEE Conf. Decis. Control*, pages 4033–4038, New Orleans, LA, Dec. 2007.

[5] S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge Univ. Press, Cambridge, UK, 2009.

[6] K. J. Åström and T. Hägglund. *PID Controllers: Theory, Design, and Tuning*. Instrum. Soc. of Amer., Research Triangle Park, NC, 2nd edition, 1995.

[7] A. G. Beccuti, S. Mariéthoz, S. Cliquennois, S. Wang, and M. Morari. Explicit model predictive control of dc-dc switched-mode power supplies with extended Kalman filtering. *IEEE Trans. Ind. Electron.*, 56(6):1864–1874, Jun. 2009.

[8] A. G. Beccuti, G. Papafotiou, R. Frasca, and M. Morari. Explicit hybrid model predictive control of the dc-dc boost converter. In *Proc. IEEE Power Electron. Spec. Conf.*, pages 2503–2509, Orlando, FL, Jun. 2007.

[9] J. Beerten, J. Verveckken, and J. Driesen. Predictive direct torque control for flux and torque ripple reduction. *IEEE Trans. Ind. Electron.*, 57(1):404–412, Jan. 2010.

[10] A. Bemporad. Efficient conversion of mixed logical dynamical systems into an equivalent piecewise affine form. *IEEE Trans. Autom. Control*, 49(5):832–838, May 2004.

[11] A. Bemporad, G. Ferrari-Trecate, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Trans. Autom. Control*, 45(10):1864–1876, Oct. 2000.

[12] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics and constraints. *Automatica*, 35(3):407–427, Mar. 1999.

[13] D. P. Bertsekas. *Convex Analysis and Optimization*. Athena Sci., Belmont, MA, 2003.

[14] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume I. Athena Sci., Belmont, MA, 3rd edition, 2005.

[15] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume II. Athena Sci., Belmont, MA, 4th edition, 2007.

[16] S. Bibian and H. Jin. High performance predictive dead-beat digital controller for dc power supplies. *IEEE Trans. Power Electron.*, 17(3):420–427, May 2002.

[17] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge Univ. Press, Cambridge, UK, 2004.

[18] S. P. Bradley, A. C. Hax, and T. L. Magnanti. *Applied Mathematical Programming*. Addison-Wesley, Reading, MA, 1977.

[19] G. S. Buja and G. B. Indri. Optimal pulsewidth modulation for feeding ac motors. *IEEE Trans. Ind. Appl.*, IA-13(1):38–44, Jan. 1977.

[20] R. Cagienard, P. Grieder, E. C. Kerrigan, and M. Morari. Move blocking strategies in receding horizon control. *J. of Process Control*, 17(6):563–570, Jul. 2007.

[21] D. Casadei, F. Profumo, G. Serra, and A. Tani. FOC and DTC: Two viable schemes for induction motors torque control. *IEEE Trans. Power Electron.*, 17(5):779–787, Sep. 2002.

[22] C. Cecati, A. Dell'Aquila, M. Liserre, and V. G. Monopoli. Design of H-bridge multilevel active rectifier for traction systems. *IEEE Trans. Ind. Appl.*, 39(5):1541–1550, Sep./Oct. 2003.

[23] C. Cecati, A. Dell'Aquila, M. Liserre, and V. G. Monopoli. A passivity-based multilevel active rectifier with adaptive compensation for traction applications. *IEEE Trans. Ind. Appl.*, 39(5):1404–1413, Sep./Oct. 2003.

[24] N. Celanovic and D. Boroyevich. A comprehensive study of neutral-point voltage balancing problem in three-level neutral-point-clamped voltage source PWM inverters. *IEEE Trans. Power Electron.*, 15(2):242–249, Mar. 2000.

[25] E. Cengelci, S. U. Sulistijo, B. O. Woo, P. Enjeti, R. Teoderescu, and F. Blaabjerg. A new medium-voltage PWM inverter topology for adjustable-speed drives. *IEEE Trans. Ind. Appl.*, 35(3):628–637, May/Jun. 1999.

[26] Y. T. Chang and Y. S. Lai. Online parameter tuning technique for predictive current-mode control operating in boundary conduction mode. *IEEE Trans. Ind. Electron.*, 56(8):3214–3221, Aug. 2009.

[27] J. Chen, A. Prodić, R. W. Erickson, and D. Maksimović. Predictive digital current programmed control. *IEEE Trans. Power Electron.*, 18(1):411–419, Jan. 2003.

[28] Z. Chen, W. Gao, J. Hu, and X. Ye. Closed-loop analysis and cascade control of a nonminimum phase boost converter. *IEEE Trans. Power Electron.*, 26(4):1237–1252, Apr. 2011.

[29] P. Correa, M. Pacas, and J. Rodríguez. Predictive torque control for inverter-fed induction machines. *IEEE Trans. Ind. Electron.*, 54(2):1073–1079, Apr. 2007.

[30] P. Cortés, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodríguez. Predictive control in power electronics and drives. *IEEE Trans. Ind. Electron.*, 55(12):4312–4324, Dec. 2008.

[31] P. Cortés, S. Kouro, B. La Rocca, R. Vargas, J. Rodríguez, J. I. León, S. Vazquez, and L. G. Franquelo. Guidelines for weighting factors design in model predictive control of power converters and drives. In *Proc. IEEE Int. Conf. Ind. Technol.*, pages 1–7, Gippsland, Australia, Feb. 2009.

[32] P. Cortés, A. Wilson, S. Kouro, J. Rodríguez, and H. Abu-Rub. Model predictive control of cascaded H-bridge multilevel inverters. *IEEE Trans. Ind. Electron.*, 57(8):2691–2699, Aug. 2010.

[33] A. Dell'Aquila, M. Liserre, V. G. Monopoli, and P. Rotondo. Overview of PI-based solutions for the control of dc buses of a single-phase H-bridge multilevel active rectifier. *IEEE Trans. Ind. Appl.*, 44(3):857–866, May/Jun. 2008.

[34] M. J. Duran, J. Prieto, F. Barrero, and S. Toral. Predictive current control of dual three-phase drives using restrained search techniques. *IEEE Trans. Ind. Electron.*, 58(8):3253–3263, Aug. 2011.

[35] R. W. Erickson, S. Ćuk, and R. D. Middlebrook. Large-signal modelling and analysis of switching regulators. In *Proc. IEEE Power Electron. Spec. Conf.*, pages 240–250, Cambridge, MA, Jun. 1982.

[36] R. W. Erickson and D. Maksimović. *Fundamentals of Power Electronics*. Kluwer Academic, Norwell, MA, 2nd edition, 2001.

[37] G. Ferrari-Trecate, D. Mignone, and M. Morari. Moving horizon estimation for hybrid systems. *IEEE Trans. Autom. Control*, 47(10):1663–1676, Oct. 2002.

[38] C. A. Floudas. *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Oxford Univ. Press, Oxford, UK, 1995.

[39] L. G. Franquelo, J. Rodríguez, J. I. Leon, S. Kouro, R. Portillo, and M. A. M. Prats. The age of multilevel converters arrives. *IEEE Ind. Electron. Mag.*, 2(2):28–39, Jun. 2008.

[40] T. Geyer. *Low Complexity Model Predictive Control in Power Electronics and Power Systems*. PhD thesis, Autom. Control Lab. ETH Zurich, Zurich, Switzerland, 2005.

[41] T. Geyer. Generalized model predictive direct torque control: Long prediction horizons and minimization of switching losses. In *Proc. IEEE Conf. Decis. Control*, pages 6799–6804, Shanghai, China, Dec. 2009.

[42] T. Geyer. Model predictive direct current control: Formulation of the stator current bounds and the concept of the switching horizon. *IEEE Ind. Appl. Mag.*, 18(2):47–59, Mar./Apr. 2012.

[43] T. Geyer and S. Mastellone. Model predictive direct torque control of a five-level ANPC converter drive system. *IEEE Trans. Ind. Appl.*, 48(5):1565–1575, Sep./Oct. 2012.

[44] T. Geyer, N. Oikonomou, G. Papafotiou, and F. D. Kieferndorf. Model predictive pulse pattern control. *IEEE Trans. Ind. Appl.*, 48(2):663–676, Mar./Apr. 2012.

[45] T. Geyer, G. Papafotiou, R. Frasca, and M. Morari. Constrained optimal control of the step-down dc-dc converter. *IEEE Trans. Power Electron.*, 23(5):2454–2464, Sep. 2008.

[46] T. Geyer, G. Papafotiou, and M. Morari. Model predictive control in power electronics: A hybrid systems approach. In *Proc. IEEE Conf. Decis. Control*, pages 5606–5611, Seville, Spain, Dec. 2005.

[47] T. Geyer, G. Papafotiou, and M. Morari. Hybrid model predictive control of the step-down dc-dc converter. *IEEE Trans. Control Syst. Technol.*, 16(6):1112–1124, Nov. 2008.

[48] T. Geyer, G. Papafotiou, and M. Morari. Model predictive direct torque control—Part I: Concept, algorithm and analysis. *IEEE Trans. Ind. Electron.*, 56(6):1894–1905, Jun. 2009.

[49] R. Goebel, R. G. Sanfelice, and A. Teel. Hybrid dynamical systems. *IEEE Control Syst. Mag.*, 29(2):28–93, Apr. 2009.

[50] O. Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge Univ. Press, Cambridge, UK, 2008.

[51] T. Gupta, R. R. Boudreaux, R. M. Nelms, and J. Y. Hung. Implementation of a fuzzy controller for dc-dc converters using an inexpensive 8-b microcontroller. *IEEE Trans. Ind. Electron.*, 44(5):661–669, Oct. 1997.

[52] P. W. Hammond. A new approach to enhance power quality for medium voltage ac drives. *IEEE Trans. Ind. Appl.*, 33(1):202–208, Jan./Feb. 1997.

[53] T. A. Henzinger. The theory of hybrid automata. In M. Inan and R. Kurshan, editors, *Verification of Digital and Hybrid Systems*, volume 170 of *NATO ASI Series F: Computer and Systems Sciences*. Springer-Verlag, 2000.

[54] T. A. Henzinger, P.-H. Ho, and H. Wong-Toi. Algorithmic analysis of nonlinear hybrid systems. *IEEE Trans. Autom. Control*, 43(4):540–554, Apr. 1998.

[55] D. G. Holmes and T. A. Lipo. *Pulse Width Modulation for Power Converters: Principles and Practice*. IEEE Press, Piscataway, NJ, 2003.

[56] J. Holtz. The representation of ac machine dynamics by complex signal flow graphs. *IEEE Trans. Ind. Electron.*, 42(3):263–271, Jun. 1995.

[57] J. Holtz and N. Oikonomou. Neutral point potential balancing algorithm at low modulation index for three-level inverter medium-voltage drives. *IEEE Trans. Ind. Appl.*, 43(3):761–768, May/Jun. 2007.

[58] J. Holtz and N. Oikonomou. Synchronous optimal pulsewidth modulation and stator flux trajectory control for medium-voltage drives. *IEEE Trans. Ind. Appl.*, 43(2):600–608, Mar./Apr. 2007.

[59] J. Holtz and N. Oikonomou. Estimation of the fundamental current in low-switching-frequency high dynamic medium voltage drives. *IEEE Trans. Ind. Appl.*, 44(5):1597–1605, Sep./Oct. 2008.

[60] H. Iman-Eini, J.-L. Schanen, S. Farhangi, and J. Roudet. A modular strategy for control and voltage balancing of cascaded H-bridge rectifiers. *IEEE Trans. Power Electron.*, 23(5):2428–2442, Sep. 2008.

[61] M. A. Johnson and M. H. Moradi. *PID Control: New Identification and Design Methods*. Springer, London, UK, 2005.

[62] A. X. Kaletsanos, I. S. Manolas, K. G. Pavlou, and S. N. Manias. Sliding mode control for cascaded H-bridge boost rectifiers. In *Proc. IEEE Int. Symp. Ind. Electron.*, pages 1070–1075, Bari, Italy, Jul. 2010.

[63] P. Karamanakos, T. Geyer, and S. Manias. Direct model predictive current control of dc-dc boost converters. In *Proc. Int. Power Electron. and Motion Control Conf. and Expo.*, pages DS2c.11–1–DS2c.11–8, Novi Sad, Serbia, Sep. 2012.

[64] P. Karamanakos, T. Geyer, and S. Manias. Direct voltage control of dc-dc boost converters using model predictive control based on enumeration. In *Proc. Int. Power Electron. and Motion Control Conf. and Expo.*, pages DS2c.10–1–DS2c.10–8, Novi Sad, Serbia, Sep. 2012.

[65] P. Karamanakos, T. Geyer, and S. Manias. Model predictive control of the interleaved dc-dc boost converter with coupled inductors. In *Proc. Eur. Power Electron. Conf.*, pages 1–10, Lille, France, Sep. 2013.

[66] P. Karamanakos, T. Geyer, and S. Manias. Direct voltage control of dc-dc boost converters using enumeration-based model predictive control. *IEEE Trans. Power Electron.*, 29(2):968–978, Feb. 2014.

[67] P. Karamanakos, G. Papafotiou, and S. Manias. Model predictive control strategies for dc-dc boost voltage conversion. In *Proc. Eur. Power Electron. Conf.*, pages 1–9, Birmingham, UK, Aug./Sep. 2011.

[68] P. Karamanakos, G. Papafotiou, and S. N. Manias. Model predictive control of the interleaved dc-dc boost converter. In *Proc. Int. Conf. Syst. Theory, Control and Comp.*, pages 1–6, Sinaia, Romania, Oct. 2011.

[69] P. Karamanakos, K. Pavlou, and S. Manias. A model predictive control strategy for the cascaded H-bridge multilevel rectifier based on enumeration. In *Proc. IEEE Ind. Electron. Conf.*, pages 5024–5029, Montreal, QC, Canada, Oct. 2012.

[70] P. Karamanakos, P. Stolze, R. Kennel, S. Manias, and T. Mouton. Variable switching point predictive torque control. In *Proc. IEEE Int. Conf. Ind. Technol.*, pages 422–427, Cape Town, South Africa, Feb. 2013.

[71] M. K. Kazimierczuk and A. Massarini. Feedforward control dynamic of dc/dc PWM boost converter. *IEEE Trans. Circuits Syst. I*, 44(2):143–149, Feb. 1997.

[72] M. K. Kazimierczuk and L. A. Starman. Dynamic performance of PWM dc/dc boost converter with input voltage feedforward control. *IEEE Trans. Circuits Syst. I*, 46(12):1473–1481, Dec. 1999.

[73] M. P. Kazmierkowski, R. Krishnan, and F. Blaabjerg. *Control in Power Electronics*. Academic Press, New York, 2002.

[74] R. Kennel, J. Rodríguez, J. Espinoza, and M. Trincado. High performance speed control methods for electrical machines: An assessment. In *Proc. IEEE Int. Conf. Ind. Technol.*, pages 1793–1799, Viña del Mar, Chile, Mar. 2010.

[75] F. Kieferndorf, P. Karamanakos, P. Bader, N. Oikonomou, and T. Geyer. Model predictive control of the internal voltages of a five-level active neutral point clamped converter. In *Proc. IEEE Energy Convers. Congr. Expo.*, pages 1676–1683, Raleigh, NC, Sep. 2012.

[76] G. S. Konstantinou and V. G. Agelidis. Performance evaluation of half-bridge cascaded multilevel converters operated with multicarrier sinusoidal PWM techniques. In *Proc. IEEE Conf. Ind. Electron. Appl.*, pages 3399–3404, Xi'an, China, May 2009.

[77] S. Kouro, P. Cortés, R. Vargas, U. Ammann, and J. Rodríguez. Model predictive control—A simple and powerful method to control power converters. *IEEE Trans. Ind. Electron.*, 56(6):1826–1838, Jun. 2009.

[78] S. Kouro, P. Lezana, M. Angulo, and J. Rodríguez. Multicarrier PWM with dc-link ripple feedforward compensation for multilevel inverters. *IEEE Trans. Power Electron.*, 23(1):52–59, Jan. 2008.

[79] P. C. Krause, O. Wasynczuk, and S. D. Sudhoff. *Analysis of Electric Machinery and Drive Systems*. Intersci. Publ. John Wiley & Sons Inc., Piscataway, NJ, 2nd edition, 2002.

[80] A. Kugi and K. Schlacher. Nonlinear H_∞ controller design for a dc-to-dc power converter. *IEEE Trans. Control Syst. Technol.*, 7(2):230–237, Mar. 1999.

[81] H. W. Kwon and S. Han. *Receding Horizon Control*. Springer-Verlag, London, UK, 2005.

[82] G. Labinaz, M. M. Bayoumi, and K. Rudie. A survey of modeling and control of hybrid systems. *Ann. Rev. of Control*, 21:79–92, 1997.

[83] A. Lamperski and A. D. Ames. On the existence of Zeno behavior in hybrid systems with non-isolated equilibria. In *Proc. IEEE Conf. Decis. Control*, pages 2776–2781, Cancun, Mexico, Dec. 2008.

[84] P. Landsmann and R. Kennel. Saliency-based sensorless predictive torque control with reduced torque ripple. *IEEE Trans. Power Electron.*, 27(10):4311–4320, Oct. 2012.

[85] J. I. Leon, S. Kouro, S. Vazquez, R. Portillo, L. G. Franquelo, J. M. Carrasco, and J. Rodríguez. Multidimensional modulation technique for cascaded multilevel converters. *IEEE Trans. Ind. Electron.*, 58(2):412–420, Feb. 2011.

[86] J. I. Leon, S. Vazquez, S. Kouro, L. G. Franquelo, J. M. Carrasco, and J. Rodríguez. Unidimensional modulation technique for cascaded multilevel converters. *IEEE Trans. Ind. Electron.*, 56(8):2981–2986, Aug. 2009.

[87] J. I. Leon, S. Vazquez, A. J. Watson, L. G. Franquelo, P. W. Wheeler, and J. M. Carrasco. Feed-forward space vector modulation for single-phase multilevel cascaded converters with any dc voltage ratio. *IEEE Trans. Ind. Electron.*, 56(2):315–325, Feb. 2009.

[88] F. H. F. Leung, P. K. S. Tam, and C. K. Li. The control of switching dc-dc converters—A general LQR problem. *IEEE Trans. Ind. Electron.*, 38(1):65–71, Feb. 1991.

[89] F. H. F. Leung, P. K. S. Tam, and C. K. Li. An improved LQR-based controller for switching dc-dc converters. *IEEE Trans. Ind. Electron.*, 40(5):521–528, Oct. 1993.

[90] D. Liberzon. *Switching in Systems and Control*. Birkhäuser, Boston, MA, 2003.

[91] B.-R. Lin, H.-H. Lu, and S.-C. Tsay. Control technique for high power factor multilevel rectifier. *IEEE Trans. Aerosp. Electron. Syst.*, 37(1):226–241, Jan. 2001.

[92] A. Linder, R. Kanchan, R. Kennel, and P. Stolze. *Model-based Predictive Control of Electric Drives*. Cuvillier Verlag, Göttingen, Germany, 2010.

[93] D. G. Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley, Reading, MA, 2nd edition, 1984.

[94] J. Lygeros, K. H. Johansson, S. N. Simić, J. Zhang, and S. S. Sastry. Dynamical properties of hybrid automata. *IEEE Trans. Autom. Control*, 48(1):2–17, Jan. 2003.

[95] J. M. Maciejowski. *Predictive Control with Constraints*. Prentice-Hall, Englewood Cliffs, NJ, 2002.

[96] P. Mattavelli, L. Rossetto, and G. Spiazzi. Small-signal analysis of dc-dc converters with sliding mode control. *IEEE Trans. Power Electron.*, 12(1):96–102, Jan. 1997.

[97] P. Mattavelli, L. Rossetto, G. Spiazzi, and P. Tenti. General-purpose fuzzy controller for dc-dc converters. *IEEE Trans. Power Electron.*, 12(1):79–86, Jan. 1997.

[98] J. Mattingley, Y. Wang, and S. Boyd. Receding horizon control: Automatic generation of high-speed solvers. *IEEE Control Syst. Mag.*, 31(3):52–65, Jun. 2011.

[99] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, Jun. 2000.

[100] H. du T. Mouton. Natural balancing of three-level neutral-point-clamped PWM inverters. *IEEE Trans. Ind. Electron.*, 49(5):1017–1025, Oct. 2002.

[101] T. A. Meynard, H. Foch, P. Thomas, J. Courault, R. Jakob, and M. Nahrstaedt. Multicell converters: Basic concepts and industry applications. *IEEE Trans. Ind. Electron.*, 49(5):955–964, Oct. 2002.

[102] H. Miranda, P. Cortés, J. I. Yuz, and J. Rodríguez. Predictive torque control of induction machines based on state-space models. *IEEE Trans. Ind. Electron.*, 56(6):1916–1924, Jun. 2009.

[103] N. Mohan, T. M. Undeland, and W. P. Robbins. *Power Electronics: Converters, Applications and Design*. Wiley, Hoboken, NJ, 3rd edition, 2003.

[104] G. E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8):114–117, Apr. 1965.

[105] F. Morel, X. Lin-Shi, J.-M. Rétif, B. Allard, and C. Buttay. A comparative study of predictive current control schemes for a permanent-magnet synchronous machine drive. *IEEE Trans. Ind. Electron.*, 56(7):2715–2728, Jul. 2009.

[106] A. Nabae, I. Takahashi, and H. Akagi. A new neutral-point-clamped PWM inverter. *IEEE Trans. Ind. Appl.*, IA-17(5):518–523, Sep./Oct. 1981.

[107] F. M. Oettmeier, J. Neely, S. Pekarek, R. DeCarlo, and K. Uthaichana. MPC of switching in a boost converter using a hybrid state model with a sliding mode observer. *IEEE Trans. Ind. Electron.*, 56(9):3453–3466, Sep. 2009.

[108] N. Oikonomou. *Control of Medium-Voltage Drives at Very Low Switching Frequency*. PhD thesis, Elektrotechnik, Informationstechnik und Medientechnik der Bergischen Universität Wuppertal, Wuppertal, Germany, 2008.

[109] N. Oikonomou, C. Gutscher, P. Karamanakos, F. Kieferndorf, and T. Geyer. Model predictive pulse pattern control for the five-level active neutral point clamped inverter. In *Proc. IEEE Energy Convers. Congr. Expo.*, pages 129–136, Raleigh, NC, Sep. 2012.

[110] G. Pannocchia and J. B. Rawlings. Disturbance models for offset-free model-predictive control. *AIChE J.*, 49(2):426–437, Feb. 2003.

[111] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, Reading, MA, 1994.

[112] G. Papafotiou, J. Kley, K. G. Papadopoulos, P. Bohren, and M. Morari. Model predictive direct torque control—Part II: Implementation and experimental evaluation. *IEEE Trans. Ind. Electron.*, 56(6):1906–1915, Jun. 2009.

[113] H. S. Patel and R. G. Hoft. Generalized techniques of harmonic elimination and voltage control in thyristor inverters: Part I—Harmonic elimination. *IEEE Trans. Ind. Appl.*, IA-9(3):310–317, May 1973.

[114] H. S. Patel and R. G. Hoft. Generalized techniques of harmonic elimination and voltage control in thyristor inverters: Part II—Voltage control techniques. *IEEE Trans. Ind. Appl.*, IA-10(5):666–673, Sep. 1974.

[115] M. Preindl and E. Schaltz. Sensorless model predictive direct current control using novel second-order PLL observer for PMSM drive systems. *IEEE Trans. Ind. Electron.*, 58(9):4087–4095, Sep. 2011.

[116] M. Preindl, E. Schaltz, and P. Thøgersen. Switching frequency reduction using model predictive direct current control for high-power voltage source inverters. *IEEE Trans. Ind. Electron.*, 58(7):2826–2835, Jul. 2011.

[117] Y. Qiu, H. Liu, and X. Chen. Digital average current-mode control of PWM dc-dc converters without current sensors. *IEEE Trans. Ind. Electron.*, 57(5):1670–1677, May 2010.

[118] J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill, Madison, WI, 2009.

[119] J. Rodríguez, S. Bernet, B. Wu, J. O. Pontt, and S. Kouro. Multilevel voltage-source-converter topologies for industrial medium-voltage drives. *IEEE Trans. Ind. Electron.*, 54(6):2930–2945, Dec. 2007.

[120] J. Rodríguez, R. M. Kennel, J. R. Espinoza, M. Trincado, C. A. Silva, and C. A. Rojas. High-performance control strategies for electrical drives: An experimental assessment. *IEEE Trans. Ind. Electron.*, 59(2):812–820, Feb. 2012.

[121] J. Rodríguez, J.-S. Lai, and F. Z. Peng. Multilevel inverters: A survey of topologies, controls, and applications. *IEEE Trans. Ind. Electron.*, 49(4):724–738, Aug. 2002.

[122] H. J. Sira-Ramírez. Nonlinear P-I controller design for switchmode dc-to-dc power converters. *IEEE Trans. Circuits Syst. I*, 38(4):410–417, Apr. 1991.

[123] E. D. Sontag. Nonlinear regulation: The piecewise linear approach. *IEEE Trans. Autom. Control*, 26(2):346–358, Apr. 1981.

[124] P. Stolze, F. Bauer, P. Landsmann, R. Kennel, and T. Mouton. Predictive torque control of an induction machine fed by a neutral point clamped inverter. In *Proc. Workshop on Pred. Control of Elect. Drives and Power Electron.*, pages 24–29, Munich, Germany, Oct. 2011.

[125] P. Stolze, P. Karamanakos, R. Kennel, S. Manias, and T. Mouton. Variable switching point predictive torque control for the three-level neutral point clamped inverter. In *Proc. Eur. Power Electron. Conf.*, pages 1–10, Lille, France, Sep. 2013.

[126] P. Stolze, M. Tomlinson, D. du Toit, R. Kennel, and T. Mouton. Predictive torque control of an induction machine fed by a flying capacitor converter. In *Proc. IEEE Conf. Africon*, pages 1–6, Livingstone, Zambia, Sep. 2011.

[127] I. Takahashi and T. Noguchi. A new quick-response and high-efficiency control strategy of an induction motor. *IEEE Trans. Ind. Appl.*, IA-22(5):820–827, Sep. 1986.

[128] S.-C. Tan, Y. M. Lai, and C. K. Tse. General design issues of sliding-mode controllers in dc-dc converters. *IEEE Trans. Ind. Electron.*, 55(3):1160–1174, Mar. 2008.

[129] L. M. Tolbert and T. G. Habetler. Novel multilevel inverter carrier-based PWM method. *IEEE Trans. Ind. Appl.*, 35(5):1098–1107, Sep./Oct. 1999.

[130] F. D. Torrisi and A. Bemporad. HYSDEL—A tool for generating computational hybrid models for analysis and synthesis problems. *IEEE Trans. Control Syst. Technol.*, 12(2):235–249, Mar. 2004.

[131] C. D. Townsend, T. J. Summers, and R. E. Betz. Multigoal heuristic model predictive control technique applied to a cascaded H-bridge StatCom. *IEEE Trans. Power Electron.*, 27(3):1191–1200, Mar. 2012.

[132] L. E. Trotter, Jr. and C. M. Shetty. An algorithm for the bounded variable integer programming problem. *J. of the Assoc. for Comput. Mach.*, 21(3):505–513, Jul. 1973.

[133] M. Vasiladiotis, K. Pavlou, S. Manias, and A. Rufer. Model predictive-based control method for cascaded H-bridge multilevel active rectifiers. In *Proc. IEEE Energy Convers. Congr. Expo.*, pages 3200–3207, Phoenix, AZ, Sep. 2011.

[134] S. Vazquez, J. I. Leon, J. M. Carrasco, L. G. Franquelo, E. Galvan, M. Reyes, J. A. Sanchez, and E. Dominguez. Analysis of the power balance in the cells of a multilevel cascaded H-bridge converter. *IEEE Trans. Ind. Electron.*, 57(7):2287–2296, Jul. 2010.

[135] A. J. Watson, P. W. Wheeler, and J. C. Clare. A complete harmonic elimination approach to dc link voltage balancing for a cascaded multilevel rectifier. *IEEE Trans. Ind. Electron.*, 54(6):2946–2953, Dec. 2007.

[136] Y. Xie, R. Ghaemi, J. Sun, and J. S. Freudenberg. Implicit model predictive control of a full bridge dc-dc converter. *IEEE Trans. Power Electron.*, 24(12):2704–2713, Dec. 2009.

[137] J. Xu, G. Zhou, and M. He. Improved digital peak voltage predictive control for switching dc-dc converters. *IEEE Trans. Ind. Electron.*, 56(8):3222–3229, Aug. 2009.

[138] P. Zanchetta, D. B. Gerry, V. G. Monopoli, J. C. Clare, and P. W. Wheeler. Predictive current control for multilevel active rectifiers with reduced switching frequency. *IEEE Trans. Ind. Electron.*, 55(1):163–172, Jan. 2008.