University of Padua
Department of Industrial Engineering

Doctoral School of Industrial Engineering
Curriculum in Energy Engineering
XXVI Cycle

Novel Model Predictive Control of a PM Synchronous Motor Drive

Design of the Innovative Structure, Feasibility and Stability Analysis, Efficient Implementation, Experimental Validation

Director of School
Ch.mo Prof. Paolo Colombo
Paolo Colombo

Coordinator of Curriculum
Ch.ma Prof.ssa Luisa Rossetto
Luisa Rossetto

Supervisor
Ch.mo Prof. Silverio Bolognani
Silverio Bolognani

Author
Matthias Preindl
Matthias Preindl

December 31, 2013

Contents

Preface

Abstract

This text focuses on advanced torque control of permanent magnet synchronous motor drives. A novel modular structure is introduced to simplify the design and implementation of *Model Predictive Control* (MPC). The layout consists of the control and the control framework. The dynamic control is the novel *virtual flux* controller, which is used to reach desired reference values, and the state observer, which is used to reduce effects of non-modeled system properties. The control framework consists of static mappings to simplify the control problem. Besides the $\alpha\beta$ and dq transformations, a reference generation procedure is used to generate state references based on optimality criteria. Also, the actuation scheme is part of the control framework and defines the available input set and the resulting control properties. The first method actuates directly switch states, i.e. voltage vectors, which yield an integer set named *Finite Control Set* (FCS). The other method actuates duty cycles via modulation, which yield the *Convex Control Set* (CCS).

A stability analysis is carried out for both, CCS-MPC and FCS-MPC. MPC is called *stable*, if it is feasible and convergent, which can be ensured using the main MPC stability theorem. However, stringent computation requirements make it difficult to apply the theorem in practice. Thus, the *Lyapunov based* MPC approach is applied to the motor drive, which provides stability guarantees independent of the *prediction horizon*. A stability constraint based on *control Lyapunov functions* (CLF) ensures convergence to the origin and the resulting optimal control problem is shown to be feasible for all time. In other words, a control input can be found at each sampling instant, which satisfies all constraints and yields a stable closed-loop system. The properties of CCS-MPC are derived using a nonlinear controller and the constrained closed-loop system is shown to be stable in the sense of Lyapunov. The stability properties of FCS-MPC are more complex due to the integer input set. Using set-theoretical methods, it is shown that a sufficiently large control error can be steered towards the origin. In other words, the proposed FCS-MPC is shown to be set stable, i.e. the control error is guaranteed to converge to a well-defined neighborhood of the origin.

MPC requires that a *Constrained Finite Time Optimal Control* (CFTOC) problem is solved at each sampling time. Small sampling periods and limited computation capabilities of embedded hardware require the CFTOC to be sufficiently simple, which is achieved using the virtual flux model in the static reference frame. The problem size is contained using a sufficiently small prediction horizon and efficient algorithms are necessary to provide a result within a sampling period. The CFTOC of the proposed CCS-MPC is a (convex) linear or quadratic programming problem, which can be solved using existing efficient algorithms. To provide a minimal approach, an efficient

algorithm is introduced to solve the one-step-ahead prediction CFTOC analytically. FCS-MPC results in a mixed integer programming problem and is therefore more difficult to solve with standard numerical methods. In practice, the CFTOC is solved by enumeration, which is combined with branch-and-bound, i.e. branch-and-cut, techniques to improve the computational efficiency.

The control algorithms have been developed on a Software-in-the-Loop (SiL) platform based on Matlab/Simulink and the code is implemented without modification on an experimental test-bench. The evaluation confirms the design and implementation of CCS-MPC and FCS-MPC and shows good results in dynamic and steady-state operation. The two MPC approaches have complimentary properties, which can be used to target different applications. CCS-MPC achieves a constant switching frequency and is a promising alternative to proportional-integral (PI) vector control. The concept can be combined with different modulation schemes, e.g. the *Symmetric Space Vector Modulation* (SSVM) and the *Discontinuous Space Vector Modulation* (DSVM) are used in this text. FCS-MPC takes the inverter switching into account and achieves an approximately constant switching ripple but a variable switching frequency. The concept is most profitably applied to systems where a high sampling frequency compared to the switching frequency is desired, e.g. high power or servo drives. Moreover, FCS-MPC lacks *Pulse Width Modulation* (PWM) harmonics in its current spectrum. Consequently, it is advantageous in terms of acoustic noise since emphasized tones are missing. However, the distinguished PWM harmonics of CCS-MPC are simpler to filter.

In summary, it can be said that the work on advanced torque control of permanent magnet synchronous motor drives has produced an innovative strategy. The introduction of a new structure has significantly simplified the model predictive control problem, the concept of stability in particular. Moreover, this structure results in the implementation of simple algorithms, which can be computed efficiently.

Kurzfassung

Der Schwerpunkt dieser Arbeit ist die Entwicklung einer fortschrittlichen Drehmomentregelung für Antriebssysteme mit einer permanentmagneterregten Synchronmaschine. Eine neue modulare Struktur wird verwendet, um die Entwicklung und Implementierung der modellbasierten prädiktiven Regelung (model predictive control, MPC) zu vereinfachen. Das Kontrollsystem besteht aus der dynamischen Regelung und dem Regelrahmen. Der dynamische Regler besteht aus einem neuen, virtuellen Flussregler, der zum Erreichen von gewünschten Sollwerten verwendet wird, und aus einem Zustandsbeobachter, der die Auswirkungen von nicht modellierten Systemeigenschaften reduziert. Der Regelrahmen besteht aus statischen Transformationen, die zur Vereinfachung des Regelproblems dienen. Neben den $\alpha\beta$ und dq Transformationen wird ein Verfahren zur Erzeugung von Zustandsreferenzen basierend auf einem Optimalitätskriterium verwendet. Darüber hinaus wird ein Verfahren zur Anwendung des Regeleingangs eingesetzt, das die Menge der verfügbaren Eingangsgrößen und die daraus resultierenden Steuereigenschaften definiert. Beim direkten Verfahren wird der Schaltzustand bzw. Spannungsvektor des Inverters ermittelt, der eine endliche Kontrollmenge (finite control set, FCS) ergibt. Im zweiten Verfahren werden Arbeitszyklen (duty-cycles) mittels Pulsweitenmodulation (pulse width modulation, PWM) angewandt, welche zu einer konvexen Kontrollmenge (convex control set, CCS) führen.

Eine Stabilitätsanalyse wird sowohl für das CCS-MPC und FCS-MPC durchgeführt. MPC wird als stabil bezeichnet, wenn es eine zulässige Lösung für das definierte Optimierungsproblem gibt und die resultierende Regelung konvergent ist. Diese Eigenschaften können mit dem grundlegenden MPC Stabilitätssatz gewährleistet werden. In der Praxis ist die Verwendung des Satzes aber aufgrund von Berechnungslimits schwierig. Deshalb wird der Lyapunov-basierte MPC (Lyapunov-based MPC) Ansatz auf Antriebssysteme angewendet. Der neue Ansatz ist vorteilhaft, weil er unabhängig von dem verwendeten Prädiktionshorizont (prediction horizon) Stabilität garantiert. Er benötigt eine Stabilitätsbedingung, die auf Lyapunov-Kontrollfunktionen (control Lyapunov functions, CLF) basiert und gewährleistet die Konvergenz zum Ursprung. Darüber hinaus wird gezeigt, dass das resultierende Optimierungsproblem immer lösbar ist. In anderen Worten, es gibt zu jedem Abtastzeitpunkt eine Eingangsgröße, die alle Bedingungen erfüllt und zu einem stabilen Regelverhalten führt. Die Eigenschaften von CCS-MPC werden von einem nichtlinearen Regler abgeleitet und die geschlossene Regelung mit Grenzwerten ist stabil im Sinne von Lyapunov. Aufgrund des nichtkontinuierlichen Aufbaus des FCS sind die Stabilitätseigenschaften von FCS-MPC komplexer. Mit einem mengentheoretischen Ansatz wird gezeigt, dass ein ausreichend großer Regelfehler in eine genau definierte Umgebung des Ursprungs gelenkt und dort gehalten werden kann.

Bei MPC wird ein Optimierungsproblem (constrained finite time optimal control, CFTOC) an jedem Abtastzeitpunkt gelöst. Damit das Problem mit Mikrocontrollern in den (kleinen) leistungselektronischen Abtastperioden gelöst werden kann, wird es im virtuellen Flussraum und statischem Referenzsystem formuliert. Die Dimension des Optimierungsproblems wird typischerweise durch kurze Prädiktionshorizonte begrenzt und effiziente Algorithmen werden benötigt, um ein Lösung innerhalb einer Abtastperiode zu erhalten. Das CFTOC von CCS-MPC ist ein (konvexes) lineares oder quadratisches Programm (linear program, lp; quadratic program, qp), das mit bereits bekannten effizienten Algorithmen lösbar ist. Um einen minimalen Ansatz vorzuführen, wird ein effizienter analytischer Algorithmus zur Lösung des CFTOC mit einem Prädiktionsschritt präsentiert. FCS-MPC resultiert in einem gemischt-ganzzahligen (mixed-integer) lp oder qp und der Lösungsweg ist dementsprechend komplex. In der Praxis werden alle zulässigen Lösungen berechnet, um daraus das Optimum zu wählen. Dieser Ansatz wird mit Branch-and-Bound bzw. Branch-and-Cut Techniken kombiniert um die numerische Effizienz zu erhöhen.

Die Regelalgorithmen wurden auf einer Software-in-the-Loop (SiL) Plattform basierend auf Matlab/Simulink entwickelt und ohne Änderung auf einen experimentellen Prüfstand übertragen. Die Evaluation bestätigt die Entwicklung und Umsetzung des CCS-MPC und FCS-MPC und zeigt gute Ergebnisse im dynamischen und stationären Betrieb. Die beiden MPC Ansätze haben unterschiedliche Eigenschaften, die in verschiedenen Anwendungen vorteilhaft sind. CCS-MPC hat eine konstante Schaltfrequenz und ist eine aussichtsreiche Alternative zur proportional-integralen (PI) Vektorregelung. Das Konzept kann mit verschiedenen Modulationsverfahren kombiniert werden, z.B. wird die symmetrische Raumzeigermodulation (Symmetric Space Vector Modulation, SSVM) und die diskontinuierliche Raumzeigermodulation (Discontinuous Space Vector Modulation, DSVM) angewendet. FCS-MPC berücksichtigt das Schaltverhalten des Inverters und erreicht eine annähernd konstante Schaltwelligkeit, aber eine variable Schaltfrequenz. Das Konzept ist interessant für Systeme, bei denen eine hohe Abtastfrequenz im Vergleich zur Schaltfrequenz erwünscht ist, z.B. Hochleistungs- oder Servoantriebe. Darüber hinaus weist FCS-MPC keine hervorgehobene PWM Oberschwingungen in seinem Spektrum auf. In Bezug auf die Geräuschentwicklung ist dies vorteilhaft, da hervorgehobene Töne fehlen und das System im Betrieb als Rauschen wahrgenommen wird. Die konzentrierten PWM Oberschwingungen von CCS-MPC sind aber einfacher zu filtern.

Abschließend kann gesagt werden, dass die Arbeit an einer fortschrittlichen Drehmomentregelung für Antriebssysteme mit einer permanentmagneterregten Synchronmaschine zu einer innovativen Strategie führte. Das Einführen einer neuen Kontrollstruktur hat das Problem der modellbasierten prädiktiven Regelung maßgeblich vereinfacht, besonders im Bereich der Stabilität. Zudem führt diese Struktur zur Implementierung einfacher Algorithmen, die effizient berechnet werden können.

Sommario

Il soggetto affrontato dal presente lavoro sono i controlli avanzati di coppia per azionamenti con un motore sincrono a magneti permanenti. A questo scopo, è stata introdotta una struttura modulare che semplifica la progettazione e l'implementazione del controllo predittivo basato su un modello (model predictive control, MPC): lo schema è costituito dal controllo dinamico e dal quadro di controllo. Il controllo dinamico è un regolatore di flusso virtuale, utilizzato per raggiungere un valore di riferimento voluto e un osservatore di stato che serve a ridurre gli effetti delle proprietà non modellizzate del sistema. Il problema del controllo è stato semplificato tramite l'utilizzo di trasformate statiche chiamate quadro di controllo. Accanto alle trasformate $\alpha\beta$ e dq viene usata una procedura per la generazione di riferimenti di stato, basati su un criterio ottimale. Il quadro di controllo contiene anche lo schema di attuazione, che serve per definire l'insieme di ingressi disponibili. Da un lato, il controllore comanda in modo diretto l'accensione e lo spegnimento dei semiconduttori, ovvero i vettori di tensione, ottenendo un insieme finito d'ingressi (Finite Control Set, FCS). Dall'altro lato vengono attuati cicli di accensione (duty-cycles) attraverso una modulazione (pulse width modulation, PWM): ciò risulta in un insieme convesso d'ingressi (convex control set, CCS).

È stata eseguita un'analisi di stabilità sia per CCS-MPC sia per FCS-MPC. MPC è stabile, se il problema di controllo ottimale ad esso associato è risolvibile e l'errore di stato converge all'origine. Tale stabilità può essere garantita attraverso il principale teorema di stabilità di MPC. Tuttavia, i requisiti di calcolo restrittivi rendono il teorema difficilmente applicabile nella pratica. Di conseguenza, viene introdotto l'approccio MPC basato su Lyapunov (Lyapunov-based MPC) per gli azionamenti, il quale fornisce garanzie sulla stabilità indipendentemente dall'orizzonte di predizione. Un vincolo di stabilità basato sulle funzioni di controllo di Lyapunov (control Lyapunov function, CLF) assicura la convergenza all'origine ed è stato provato che il problema ottimale di controllo risultante è sempre risolvibile. In altre parole, ad ogni istante di campionamento si può trovare un ingresso che soddisfi tutti i vincoli del sistema e renda stabile il sistema a circuito chiuso. Le proprietà di CCS-MPC vengono ottenute utilizzando un controllo non lineare ed è dimostrato che il sistema vincolato ad anello chiuso è stabile secondo Lyapunov. Le proprietà di stabilità di FCS-MPC sono più complesse a causa dell'insieme non continuo d'ingressi. Utilizzando metodi della teoria degli insiemi si dimostra che un errore di controllo sufficientemente ampio può essere diretto verso l'origine e tenuto in un dintorno dell'origine ben definito.

MPC richiede che in ogni istante di campionamento si risolva un problema di ottimizzazione (constrained finite time optimal control, CFTOC). La limitata potenza

di calcolo dei microcontrollori e la brevità dei periodi di campionamento richiedono un CFTOC relativamente semplice, che si può ottenere utilizzando un modello di flusso virtuale nel sistema statico di riferimento. Scegliendo piccoli orizzonti di predizione si limita la dimensione del CFTOC, la cui risoluzione necessità di algoritmi efficienti, che permettano di ottenere un risultato all'interno di un periodo di campionamento. Il CFTOC di CCS-MPC è un programma (convesso) lineare o quadratico (linear program, lp; quadratic program, qp) che può essere risolto tramite algoritmi efficienti e noti. Al fine di elaborare una strategia di tipo minimalista, viene introdotto un algoritmo efficiente che risolve analiticamente il problema con un orizzonte di predizione di un passo. Il CFTOC di FCS-MPC è un problema di programmazione lineare o quadratico a numeri misti interi (mixed-integer) ed è quindi più difficile da risolvere con metodi numerici standard. In pratica si calcolano tutte le soluzioni possibili, tra le quali viene scelta la soluzione ottimale. Per migliorare l'efficienza di calcolo si combina quest'approccio con tecniche branch-and-bound e branch-and-cut.

Gli algoritmi di controllo sono stati sviluppati su una piattaforma software-in-the-loop (SiL) basata su Matlab/Simulink e il codice di programmazione è stato implementato su un banco di prova sperimentale, senza modifiche. La valutazione approva la progettazione e la realizzazione di CCS-MPC e FCS-MPC e indica buoni risultati sia nell'operazione dinamica che in quella stazionaria. I due approcci MPC hanno proprietà diverse che risultano vantaggiose per applicazioni differenti. CCS-MPC ha una frequenza di commutazione costante ed è un'alternativa promettente al controllo vettoriale proporzionale-integrale (PI). Il concetto può essere combinato con diversi schemi di modulazione, nella fattispecie si usa la modulazione simmetrica di vettori spaziali (symmetric space vector modulation, SSVM) e la modulazione discontinua di vettori spaziali (discontinuous space vector modulation, DSVM). FCS-MPC tiene conto della commutazione dell'inverter e raggiunge all'incirca un ripple di commutazione costante, ma ottiene una frequenza di commutazione variabile. Il concetto è vantaggioso per sistemi dove è richiesta un frequenza di campionamento alta rispetto alla frequenza di commutazione, per esempio azionamenti ad alta potenza o servoazionamenti. Inoltre, lo spettro della corrente di FCS-MPC non contiene armoniche PWM e di conseguenza è vantaggioso in termini di rumore acustico, data la mancanza di toni distinti. Tuttavia, le armoniche PWM distinte di CCS-MPC sono più semplici da filtrare.

Si può concludere affermando che lo studio del problema dei controlli avanzati di coppia per azionamenti con un motore sincrono a magneti permanenti, ha portato all'individuazione di una strategia innovativa. L'introduzione di una nuova struttura di controllo ha semplificato notevolmente il problema di controllo predittivo, con particolare attenzione al concetto di stabilità. Inoltre, le implementazioni di tale struttura si sono rivelate particolarmente efficaci su piano computazionale.

Part I.

Permanent Magnet Synchronous Machine Drive System

Introduction

A *drive system* converts electrical into mechanical energy or vice-versa. It consist of three major parts: the electrical machine, the power electronic converter, and the control system, which is shown in Figure I.1. The *electrical machine* provides or absorbs mechanical energy in form of torque applied to a rotating shaft to or from an external mechanical system called *load*. The machine is nominated motor or generator with respect to the primary energy flow [2], [11], [33]. The motor convention is used, where the energy flow is positive if the machine absorbs electrical energy and produces mechanical energy. The electromechanical energy transformation happens according to the law of conservation of energy. During the process, energy is stored primarily in form of magnetic energy due to currents circulating in inductive circuits and angular kinetic energy due to the rotating shaft. Both magnetic and kinetic energy storage is required to operate a system at a given operation point and the stored energy is returned when the system is stopped. Moreover, electrical and mechanical energy is dissipated, i.e. transformed into heat. They are classified into copper (Joule, proximity, skin effect), iron (hysteresis and eddy-current), and mechanical friction losses [2], [11], [33].

Electrical machines are distinguished by whether they use direct-current (DC) or alternating-current (AC) with two, three, or more phases. In contrast to DC machines, AC machines have a simpler setup (without commutator) leading to an increased reliability and compactness [2], [33]. Classic AC machines have a three-phase stator layout and the rotor defines whether the machine is based on the synchronous machine or induction (or asynchronous) machine principle, both featuring an armature and field winding. The *field winding* produces a rotating magnetic field and is located at the rotor of AC machines. The three phase *armature winding* is located at the stator and produces current, which rotates spatially at the same velocity as the rotor flux. The interaction of the armature current and the magnetic field flux produces torque. In this text, the focus is on the *Permanent Magnet Synchronous Machine* (PMSM), where the field coils are replaced by permanent magnets. The PMSM gained popularity with the development of high performance permanent magnets, mainly the rare-earth magnets, i.e. the Neodymium (NeFeB) and Samarium-Cobalt (SmCo) magnets. These permanent magnets allow a compact rotor design and the resulting machines have a particularly high torque density [13]. Moreover, the PMSM

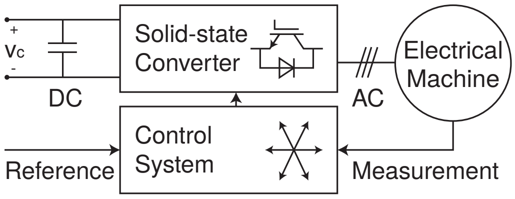


Figure I.1.: Drive system

does not require a current to produce a magnetic field leading to advantages in terms of efficiency [13].

The PMSM is fed by a solid-state power converter, which transforms energy from DC into AC and vice-versa. Similar to electric machines, AC/DC converters are named inverter or rectifier with respect to the primary energy flow. According to the motor convention, the converter is named inverter in this text. The system is fed by a DC link and is typically connected to a larger electrical system, e.g. the low or medium voltage power grid or a distributed DC bus. In this text, it is assumed that the DC link is provided by an external system, which can supply and absorb electric energy without significant effect. In particular, a *Voltage Source Inverter* (VSI) [65] is used that requires an approximately constant DC link voltage and is supported by a DC link capacitor. The connection of the AC and DC side is established by power electronic devices [65], e.g. MOSFETs and IGBTs, which are used as electric valves. The connection scheme of the valves defines the inverter topology and its operation principle. Several topologies are available, e.g. the two level voltage source inverter (VSI), multilevel VSI [70], and modular multilevel inverter [23]. The two level VSI has become a de-facto standard for research and industry in many applications due to its simplicity and ruggedness and is used in this research.

A VSI applies a voltage with variable frequency and magnitude to the terminals of an inductive AC device, i.e. the three-phase PMSM. This voltage is chosen by a control system, which ensures that the drive system behaves in the expected manner. The control goal is to apply a desired torque to a rotating shaft. The torque reference can be generated locally or communicated via fieldbus, e.g. CAN or EtherCat. It is set by a higher control layer with the goal to control a larger system, e.g. control of a wind turbine at optimal efficiency or a vehicle speed controller. Based on measurements, a control error is established and the terminal voltage is set such that the control error decreases. Using advanced control techniques, it is possible to integrate closed loop torque control into a single block [43], [81], [82], [97] (or even closed loop speed control [69], [80]). However, the resulting control problem is generally difficult to solve and a common alternative is to break the problem down into subproblems. Blocks with static mappings named *control framework* are introduced to reduce complexity.

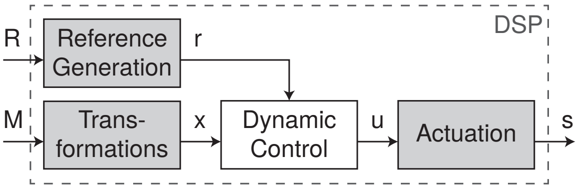


Figure I.2.: Control framework (gray) with measurements M, states x, reference torque R, reference states r, plant input u, and switching function s

They are outlined in Figure I.2. It includes transformations between the flux and current space in the three-phase, $\alpha\beta$ and dq reference frame. Moreover, the *reference generation* procedure is used to generate state reference vector from a torque reference value. The state reference is computed based on an optimality criteria, which unifies *Maximum Torque per Ampere* (MTPA) tracking and *Field Weakening* (FW) in a single block. Also, the *actuation* scheme is part of the control framework. There the inverter duty cycles or switch states are computed such that the inverter actuates the desired terminal voltage. The *dynamic control* considers the evolution of the control error over time and is described in Part II of this text.

Part I of this text is divided into three chapters. In Chapter 1, the operating principle of the PMSM is described pointing out the rotor stator interaction. Different PMSM machine designs are shown and nonlinear effects, i.e. saturation and cross-saturation, are explained. State-space systems are developed to describe the electrical dynamics in discrete time since modern control is almost exclusively implemented on sampled digital hardware. The state of the dynamic models is the stator flux, which leads to simple linear formulations and is obtained from the current measurements using a flux-current map. The chapter is concluded introducing the PMSM constraints, which define the available steady-state operation points. In Chapter 2, the torque generation of the PMSM is explained. Since the current, i.e. flux, which produces a desired torque is not unique, the operation point is chosen in order to maximize the efficiency. The related optimization problem is defined and an efficient procedure to solve the problem are presented. The chapter is concluded by addressing the model inconsistency between the model using constant parameters and the real machine featuring nonlinear effects, e.g. saturation. Since the PMSM model based on the linear flux-current relationship is an approximation, a strategy is proposed to improve the local model correspondence. In Chapter 3, the operating principle of the inverter is shown. The concept of switching function and duty cycle as well as model nonlinear inverter effects are introduced. Moreover, the interface between the gate drivers of the solid-state valves and the control algorithm is defined. A control algorithm stabilizes a desired terminal voltage and the so-called actuation block (direct actuation or modulation) generates a sequence of desired gate signals.

Chapter 1.

Electromagnetic Field and Model

In this chapter, the operating principle of the PMSM is described pointing out the rotor stator interaction. Different PMSM machine designs are shown and nonlinear effects, i.e. saturation and cross-saturation, are explained. The electrical dynamics of the PMSM are described using linear state-space systems. Since modern control is almost exclusively implemented on sampled digital hardware, the systems are transformed into discrete time. Moreover, the constraints are shown, where the PMSM can be operated in steady-state conditions.

1.1. Armature Winding

The dynamic of electrical machines is typically described as systems of differential equations [54]. These equations are based on the electric, magnetic, and mechanic effects. In this section, the physical behavior of the PMSM is illustrated. This description is the basis for writing a control model of the PMSM.

A stationary conductor placed in a time-varying magnetic field is subject to electromagnetic induction. An example is shown in Figure 1.1. This phenomena is described by Faraday's law of induction [38], which states

$$e_w(t) = \dot{\lambda}_w(t), \tag{1.1}$$

where $e_w(t) \in \mathbb{R}$ is the electric motion force and $\lambda_w(t)$ is the flux linkage. The flux linkage is the amount of flux acting on the conductor. In a winding with $N_w \in \mathbb{N}$ loops of a thin wire in an uniform magnetic field with flux $\varphi_w(t)$, the flux linkage is $\lambda_w(t) = N_w \varphi_w(t)$, where $\varphi_w(t) \in \mathbb{R}$ is the magnetic flux in the cross section of the winding. At the terminals of the winding, the voltage [54], [65]

$$v_w(t) = R_w i_w(t) + e_w(t) = R_w i_w(t) + \dot{\lambda}_w(t), \tag{1.2}$$

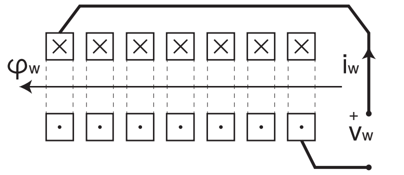


Figure 1.1.: Winding in magnetic field

is measured where $R_w i_w(t)$ is the resistive voltage drop. The parameter $R_w \in \mathbb{R}_+$ is the electrical resistance and $i_w(t)$ is the current flowing in the winding.

A three-phase winding is obtained by placing three windings at an electrical angle of $2\pi/3$ as shown in Figure 1.2. This assembly can be repeated p times along a circumference, where $p \in \mathbb{N}$ is the number of pole pairs. Thus, the mechanical angle between windings is $2\pi/(3p)$. The stator assembly of an electrical machine with radial magnetic field and internal rotor is shown in Figure 1.2(a). The armature winding consists of three coils as depicted in Figure 1.2(b). The coils are connected to each other in order to form a three-phase circuit and the windings are cabled such that they form a star connection as portrayed in Figure 1.2(b). If the connection forms a triangle, it can be treated by computing the equivalent star [2].

At the machine terminals, the voltage

$$v_{ph,0}(t) = \begin{bmatrix} v_{1,0}(t) \\ v_{2,0}(t) \\ v_{3,0}(t) \end{bmatrix} = \begin{bmatrix} v_1(t) \\ v_2(t) \\ v_3(t) \end{bmatrix} + \mathbf{1} v_0(t) = v_{ph}(t) + \mathbf{1} v_0(t) \in \mathbb{R}^3 \tag{1.3}$$

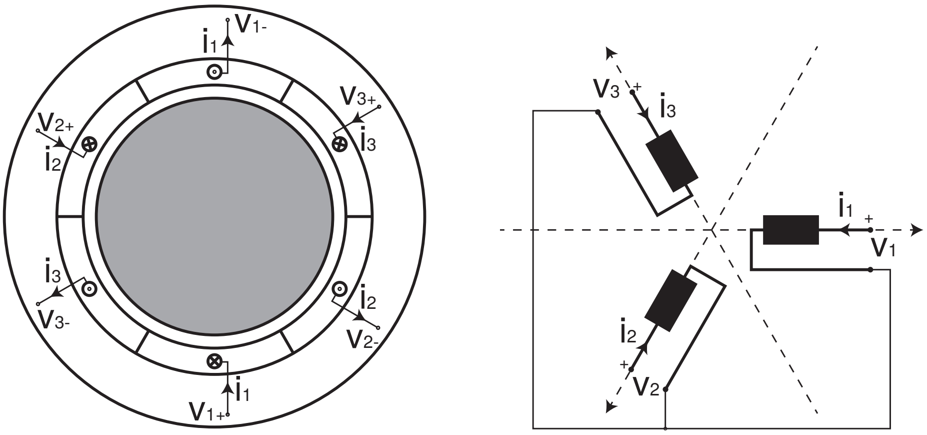
is applied, where $\mathbf{1}$ is the vector of ones with appropriate dimension. The voltage $v_{ph,0}(t) \in \mathbb{R}^3$ has two components. The voltage $v_{ph}(t) \in \mathbb{R}^3$ is the (difference) voltage, which is applied to each phase winding. The (zero) voltage $v_0(t) \in \mathbb{R}$ is the voltage, which is applied between the neutral point of the electrical machine and the neutral potential of the drive system. Assuming that the windings are subject to a flux linkage $\lambda_{ph}(t) = [\lambda_1(t), \lambda_2(t), \lambda_3(t)]' \in \mathbb{R}^3$, the stator equations are

$$v_{ph}(t) = v_{ph,0}(t) - \mathbf{1} v_0(t) = R_s i_{ph}(t) + \dot{\lambda}_{ph}(t), \tag{1.4}$$

where $R_s i_{ph}(t)$ is the resistive voltage drop. The parameter $R_s \in \mathbb{R}_+$ is the resistance per phase assuming that each winding is equal and $i_{ph}(t) = [i_1(t), i_2(t), i_3(t)]' \in \mathbb{R}^3$ is the current per phase.

To simplify the analysis, the equation is transformed into the $\alpha\beta$ reference frame. Details on the Clarke transformation $\mathbf{T}_{\alpha\beta}$ are treated in Appendix A. Using this $\alpha\beta$ transformation yields

$$v_{\alpha\beta}(t) = R_s i_{\alpha\beta}(t) + \dot{\lambda}_{\alpha\beta}(t), \tag{1.5}$$



(a) Assembly

(b) Concept

Figure 1.2.: Three-phase armature winding

with the $\alpha\beta$ quantities: the terminal voltage $v_{\alpha\beta}(t) = \mathbf{T}_{\alpha\beta} v_{ph}(t) = \mathbf{T}_{\alpha\beta} v_{ph,0}(t) \in \mathbb{R}^2$, the phase current $i_{\alpha\beta}(t) = \mathbf{T}_{\alpha\beta} i_{ph}(t) \in \mathbb{R}^2$, and the flux linkage $\lambda_{\alpha\beta}(t) = \mathbf{T}_{\alpha\beta} \lambda_{ph}(t) \in \mathbb{R}^2$.

Now, the zero component of the system is considered. For neutral point isolate machines, the zero component of the phase current $i_0(t) = \frac{1}{3}\mathbf{1}' i_{ph}(t) \equiv 0$ by the law of Kirchhoff. Moreover, $v_0(t) = \frac{1}{3}\mathbf{1}' v_{ph,0}(t)$ and consequently $\frac{1}{3}\mathbf{1}' v_{ph}(t) = 0$ and $\frac{1}{3}\mathbf{1}' \lambda_{ph,0}(t) = 0$. In other words, a zero voltage applied to the PMSM terminals results in a voltage between the neutral point of the machine and the neutral potential of the drive system. Consequently, no voltage is applied to the windings nor can it vary the machine flux. Moreover, a zero voltage applied at the machine terminals cannot produce torque since it cannot drive currents nor flux in the machine. Thus, the zero system is generally neglected when analyzing and modeling an electrical machine.

A three-phase electrical machine with rotating field is often described in the *dq* reference frame. Details on the Park transformation $\mathbf{T}_{dq}(\epsilon(t))$ are treated in Appendix A. Using this *dq* transformation, the voltage equation becomes

$$v_{dq}(t) = R_s i_{dq}(t) + \dot{\lambda}_{dq}(t) + \omega(t)\mathbf{J}\lambda_{dq}(t), \tag{1.6}$$

with the *dq* quantities; the terminal voltage $v_{dq}(t) = \mathbf{T}_{dq}(\epsilon(t)) v_{\alpha\beta}(t) \in \mathbb{R}^2$, the phase current $i_{dq}(t) = \mathbf{T}_{dq}(\epsilon(t)) i_{\alpha\beta}(t) \in \mathbb{R}^2$, and the derivative of the flux linkage $\dot{\lambda}_{dq}(t) = \mathbf{T}_{dq}(\epsilon(t)) \left(\dot{\lambda}_{\alpha\beta}(t) - \omega(t)\mathbf{J}\lambda_{\alpha\beta}(t) \right) \in \mathbb{R}^2$. The angular velocity $\omega(t) \in \mathbb{R}$ is the rotational velocity of the *dq* reference frame with respect to the stationary $\alpha\beta$

reference frame such that $\dot{\epsilon}(t) = \omega(t)$ and $\mathbf{J}$ is the rotation matrix $\mathbf{J} = [[0, -1]', [1, 0]']'$. Equation (1.6) can also be written in non-matrix form

$$v_d(t) = R_s i_d(t) + \dot{\lambda}_d(t) - \omega(t)\lambda_q(t), \tag{1.7a}$$

$$v_q(t) = R_s i_q(t) + \dot{\lambda}_q(t) + \omega(t)\lambda_d(t). \tag{1.7b}$$

In these equations, the phase currents $i_{dq}(t)$, the flux linkage $\lambda_{dq}(t)$, and the terminal voltage $v_{dq}(t)$ appear to be decoupled from each other. The dependencies among these quantities are introduced in the next section.

1.2. Magnetic Field

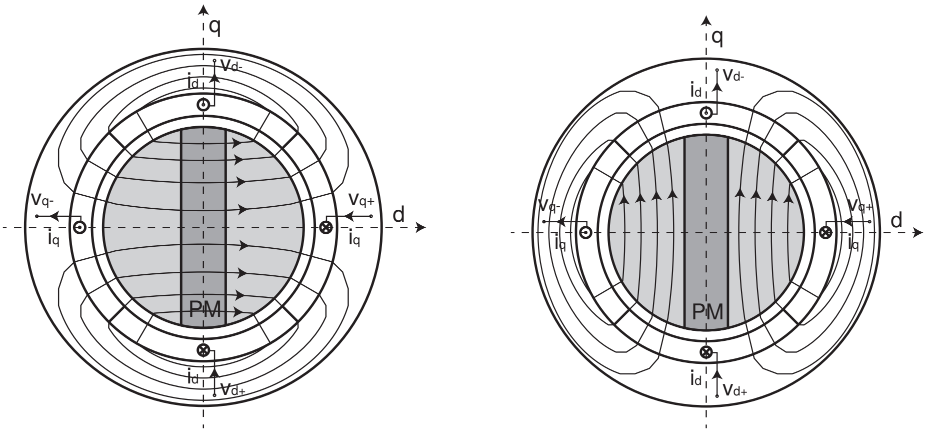
In this section, the relation between the phase current and the stator flux is studied. Assuming (initially) linear dependencies, the magnetic field of an electric machine is understood as superposition of the field generated by the rotor and the field generated by the stator [2], [13], [54]. In this section, the flux generation is analyzed by setting to zero the flux generated by the permanent magnets (PM) on the rotor and computing the flux generated by the stator currents and vice versa. The total flux results from the superposition principle. To simplify the treatment, an equivalent dq winding is used, as shown in Figure 1.3. By convention of synchronous machines, the d axis is aligned with the flux generated by the rotor, i.e. the permanent magnet flux, with angle $\epsilon(t)$. Thus, the dq winding rotates synchronously with the electrical rotor position $\epsilon(t)$ at velocity $\omega(t) = \dot{\epsilon}(t)$. The electric dynamic of this winding is described by (1.6).

First, the flux generated by the PM is assumed to be zero. PM materials have a magnetic permeability proximal to that of air. Thus, the rotor can be imagined with the same geometry but with the PM's removed. In this configuration, a d axis current produces a d axis flux as shown in Figure 1.3(a). Similarly, a q axis current produces a q axis flux as depicted in Fig 1.3(b). The flux produced by the stator currents is

$$\varphi_{dq}(t) = \begin{bmatrix} \varphi_d(t) \\ \varphi_q(t) \end{bmatrix} = \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix} \begin{bmatrix} i_d(t) \\ i_q(t) \end{bmatrix} = \mathbf{L} i_{dq}(t), \tag{1.8}$$

where the parameters $L_d \in \mathbb{R}_+$ and $L_q \in \mathbb{R}_+$ are the auto-inductance of the d and q axis, respectively.

Then, the flux generated by the currents is assumed to be zero, achieved by setting $i(t) = 0$ due to (1.8). In these conditions, the machine flux is produced only by the



(a) Flux λ_d

(b) Flux λ_q

Figure 1.3.: Flux in the PMSM with dq winding for $p = 1$

PM's. This flux is aligned with the d axis by definition of the dq reference frame. As a result, there is no flux on the q axis, which yields

$$\psi_{dq} = \begin{bmatrix} \psi \\ 0 \end{bmatrix}, \tag{1.9}$$

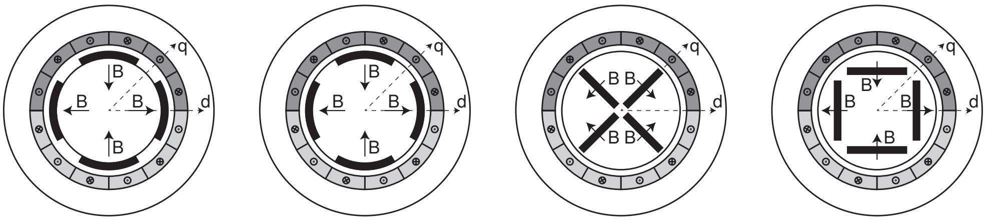
where ψ_{dq}, i.e. $\psi \in \mathbb{R}_+$, is the flux generated by the PM. The flux ψ_{dq} is constant since a PM produces a constant magnetic motion force (mmf) and the reluctance path (along the d axis) is constant as well.

The total flux of the PMSM is obtained by superposition

$$\lambda_{dq}(t) = \varphi_{dq}(t) + \psi_{dq} = \mathbf{L}i_{dq}(t) + \psi_{dq}. \tag{1.10}$$

The inductance values depend on the rotor geometry, which defines the magnetic path along the d and q axis. The inductance L_d and L_q can differ significantly and L_q/L_d is called the salience ratio. Typically, a high salience ratio, i.e. $L_d \ll L_q$, is achieved in applications, where field weakening is important [10], [92]. However, many modern machines feature a salience $L_d < L_q$ since it leads to an increased torque density [5], [6], [12], [13]. Moreover, surface mounted and interior PMSM typically satisfy $L_d \leq L_q$ [10].

Several rotor configurations have been proposed in literature [92]. Some examples are shown in Figure 1.4. The surface-mounted PMSM, see Figure 1.4(a), has a magnetically isotropic rotor, which yields $L_d = L_q$. The machine is typically not suitable for field-weakening but is simple to manufacture. The spoke PMSM



(a) Surface PM (b) Inset PM (c) Spoke PM (d) Barrier PM

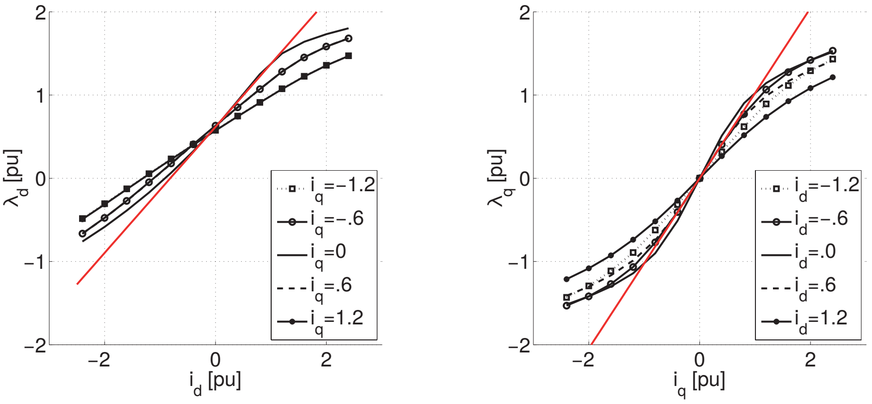
Figure 1.4.: PMSM examples with different rotor configurations for $p = 2$

configuration, see Figure 1.4(c), uses the flux concentration principle. This concept is advantageous for achieving a particularly high air-gap induction or for employing weaker, i.e. cheaper, magnets. Machines with this rotor configuration have typically a small salience. A high salience ratio is achieved by the inset PMSM, see Figure 1.4(b), and the barrier PMSM, see Figure 1.4(d). The latter one uses interior PM's, which act as flux barriers. If the salience ratio is not sufficient, multiple barriers can be inserted but the machine becomes increasingly difficult to manufacture.

Two special cases can be defined with respect to the rotor configuration. The first case is the PMSM with magnetically isotropic rotor, which leads to $L_d = L_q$. The most popular example is the surface-mounted PM rotor in Figure 1.4(a) such that surface mounted PMSM is sometimes used as acronym for isotropic PMSM. However, machines with other rotor types can be isotropic in general. The counterpart of the isotropic PMSM is a machine which produces only reluctance torque. Strictly speaking, this machine is not a PMSM since it does not feature permanent magnets. However, the reluctance machine is obtained imposing $\psi \equiv 0$. For producing a significant amount of torque, the reluctance machine requires a large salience ratio, i.e. $L_d \ll L_q$.

1.3. Saturation

In the previous section, the PMSM flux-current dependence has been introduced assuming linear dependencies. However, this model is typically an approximation since magnetic materials are used close to saturation for achieving compact designs [11]. In this operation region, the behavior is nonlinear. Literature [17], [38], [46], [56], [62], [71], [73], [90], [94]–[96] typically distinct two phenomena: saturation and cross-saturation.



(a) d-axis

(b) q-axis

Figure 1.5.: Flux-current characteristic in presence of cross-saturation (measured) and linearized model

Saturation refers to the effect, where the current saturates the magnetic material on the same (d or q) axis. Thus, the flux-current dependency is written as the nonlinear function

$$\lambda_{dq}(t) = l(i_{dq}(t)) = \begin{bmatrix} l_d(i_d(t)) \\ l_q(i_q(t)) \end{bmatrix}; \quad l : \mathbb{R}^2 \to \mathbb{R}^2 = \begin{bmatrix} l_d : \mathbb{R} \to \mathbb{R} \\ l_q : \mathbb{R} \to \mathbb{R} \end{bmatrix}. \tag{1.11}$$

Cross-saturation refers to the effect, where the current does not only saturate the material of the same axis but leads to variations of the flux of the other axis as well. It occurs when parts of the machine significantly saturate and the flux partially moves to paths of the other axis. This tendency introduces a magnetic coupling of the d and q axis. It can be modeled similarly to saturation, the flux-current dependency is written as a nonlinear function

$$\lambda_{dq}(t) = l(i_{dq}(t)) = \begin{bmatrix} l_d(i_{dq}(t)) \\ l_q(i_{dq}(t)) \end{bmatrix}; \quad l : \mathbb{R}^2 \to \mathbb{R}^2 = \begin{bmatrix} l_d : \mathbb{R}^2 \to \mathbb{R} \\ l_q : \mathbb{R}^2 \to \mathbb{R} \end{bmatrix}. \tag{1.12}$$

The cross-saturation model (1.12) is defined by two surfaces and implies the saturation model (1.11), which is defined by two curves. An example is shown in Figure 1.5, where the measured flux-current characteristic is compared to the linearized one.

Although a cross-saturation model is necessary to describe the global flux-current relation in general, introducing a cross-saturation model leads to a significant increase of complexity of analysis and design of PMSM control systems. This is due to the nonlinearity of the PMSM model per definition. In most cases, the increase in complexity is not justified since the machine behaves locally according to $\lambda_{dq} = \mathbf{L}i_{dq} + \psi_{dq}$ with good approximation. Moreover, the dependence $\lambda_{dq}(t) = l(i_{dq}(t))$ is not available for most PMSM machines and needs to be established in laboratory conditions, during commissioning, or operation [21], [22]. Thus, saturation and cross-saturation is neglected for analysis and design of PMSM control systems and the following assumption is made.

Assumption 1.1. The linear relation $\lambda_{dq} = \mathbf{L}i_{dq} + \psi dq$ is suitable as local approximation of $\lambda_{dq}(t) = l(i_{dq}(t))$ in the region of desired operation points.

Further details on this assumption are shown in Section 2.5. There, methods to optimize the parameters $\mathbf{L}$, ψ_{dq}, etc. with respect to the torque equation are shown.

1.4. State-Space Model

1.4.1. Continuous Time

First, the PMSM stator equations are written in *continuous-time* state-space form to describe the electric dynamics of the machine. The state of the PMSM is the three-phase current i_{ph} or flux λ_{ph}, dependent on the formulation. However, i_{ph} and λ_{ph} of the neutral point isolated machine have the nullspace $\mathbf{1}'$. In other words, the system is known to have a redundant dimension since $1' i_{ph} \equiv 0$, i.e. $1' \lambda_{ph} \equiv 0$. Thus, the dimension of the system can be safely reduced neglecting the zero system, which is not necessary to fully describe the system in nonfault conditions. The state becomes the Clarke and Park transformed flux or current without loss of generality.

Now, state-space models with respect to $\lambda_{\alpha\beta}$, λ_{dq}, and i_{dq}, are introduced. From (1.5), a state-space model with the state $\lambda_{\alpha\beta}$ is obtained rearranging the equation

$$\dot{\lambda}_{\alpha\beta}(t) = \bar{v}_{\alpha\beta}(t). \tag{1.13}$$

The input is $\bar{v}_{\alpha\beta}(t) = v_{\alpha\beta}(t) - R_s i_{\alpha\beta}(t)$, which is named the *compensated terminal voltage* in the $\alpha\beta$ reference frame, where $R_s i_{\alpha\beta}(t)$ is the resistive voltage drop and $R_s \in \mathbb{R}_+$ is the stator resistance. Using $\bar{v}_{\alpha\beta}(t)$ means compensating the resistive voltage drop at each time instant with the terminal voltage $v_{\alpha\beta}(t)$. This approach is convenient since $R_s i_{\alpha\beta}(t)$ is typically small compared to the available terminal voltage.

The state-space system (1.13) is linear time invariant (LTI) and defines an integrator dynamics. Consequently, the undriven ($\bar{v}_{\alpha\beta}(t) \equiv 0$) autonomous system is marginally stable since the poles of the system lie on the origin of the complex plane. The advantage of the state-space model (1.13) is its simplicity. On the downside, the flux $\lambda_{\alpha\beta}$ is an AC value, which is time-varying (sinusoidal) even in steady-state conditions. As a result, the system is not straight-forward to analyze and additional complexity is introduced in control design, e.g. steady-state offsets cannot be (completely) removed using simple integration.

A convenient alternative is a state-space model with the *dq* flux λ_{dq} as state, which is a DC (constant) value in steady-state conditions. This model is derived from (1.6)

$$\dot{\lambda}_{dq}(t) = \mathbf{A}_c(t)\lambda_{dq}(t) + \mathbf{B}_c\bar{v}_{dq}(t), \tag{1.14}$$

with the input $\bar{v}_{dq}(t) = v_{dq}(t) - R_s i_{dq}(t)$, which is the compensated voltage in the *dq* reference frame and the matrices are

$$\mathbf{A}_c(t) = -\omega(t)\mathbf{J} = \begin{bmatrix} & \omega(t) \\ -\omega(t) & \end{bmatrix}; \quad \mathbf{B}_c = \mathbf{I} = \begin{bmatrix} 1 & \\ & 1 \end{bmatrix}. \tag{1.15}$$

The state-space system (1.14) is linear time variant (LTV) and depends on the parameter $\omega(t)$. The stability properties of the undriven ($\bar{v}_{dq}(t) \equiv 0$) autonomous system are shown as follows.

Proposition 1.1. *The undriven autonomous system (1.14) is marginally stable for all $\omega(t) \in \mathbb{R}$.*

Proof. Let $\Gamma(t) \overset{\text{def}}{=} 0.5\lambda'_{dq}(t)\lambda_{dq}(t)$ be a candidate Lyapunov function ($\Gamma(t) > 0$ for all $\lambda_{dq}(t) \neq 0$; $\Gamma(t) = 0$ if $\lambda_{dq}(t) = 0$; $\Gamma(t)$ is radially unbounded and continuous in the origin). Deriving $\Gamma(t)$ with respect to time yields

$$\dot{\Gamma}(t) = \lambda'_{dq}(t)\dot{\lambda}_{dq}(t) = \lambda'_{dq}(t)\mathbf{A}_c(t)\lambda_{dq}(t) = -\omega(t)\lambda'_{dq}(t)\mathbf{J}\lambda_{dq}(t) = 0, \tag{1.16}$$

for all $\omega(t) \in \mathbb{R}$. Since the $\dot{\Gamma}(t)$ is nonpositive, the system is stable. However, the system is not asymptotically stable but marginally stable, since $\dot{\Gamma}(t)$ does not strictly belong to the negative halfplane[1]. □

[1] The evolution of the undriven autonomous system (1.14) is

$$\lambda_{dq}(t) = \begin{bmatrix} \cos\omega(t)t & \sin\omega(t)t \\ -\sin\omega(t)t & \cos\omega(t)t \end{bmatrix} \lambda_{dq}(0) = T_{dq}(\omega(t)t)\lambda_{dq}(0) \quad \forall t \in [0, \infty].$$

The state-space model (1.14) is often modified introducing the linear dependency $\lambda_{dq}(t) = \mathbf{L}i_{dq} + \psi_{dq}$. Thus, the state is linearly transformed and becomes the dq current i_{dq}, which yields

$$\dot{i}_{dq}(t) = \bar{\mathbf{A}}_c(t) i_{dq}(t) + \bar{\mathbf{B}}_c v_{dq}(t) + \bar{E}_c(t). \tag{1.17}$$

where the input $v_{dq}(t)$ is the terminal voltage in the dq reference frame. Similar to (1.13) and (1.14), the model can be written using the compensated terminal voltage $\bar{v}_{dq}(t)$ as input. The matrices are

$$\begin{aligned} \bar{\mathbf{A}}_c(t) &= -\mathbf{L}^{-1}(R_s\mathbf{I} + \omega(t)\mathbf{J}\mathbf{L}) = \begin{bmatrix} -\frac{R_s}{L_d} & \omega(t)\frac{L_q}{L_d} \\ -\omega(t)\frac{L_d}{L_q} & -\frac{R_s}{L_q} \end{bmatrix}; \\ \bar{\mathbf{B}}_c &= \mathbf{L}^{-1} = \begin{bmatrix} \frac{1}{L_d} & \\ & \frac{1}{L_q} \end{bmatrix}; \quad \bar{E}_c(t) = -\omega(t)\mathbf{L}^{-1}\mathbf{J}\psi_{dq} = \begin{bmatrix} \\ -\omega(t)\frac{\psi}{L_q} \end{bmatrix}. \end{aligned} \tag{1.18}$$

The advantage of this model is that the stator resistance becomes a part of the parameter matrix $\bar{A}_c$. On the downside, the modified dynamic increases the complexity and the model depends on several parameters. Moreover, a new offset $\bar{E}_c$ is introduced. Compared to the offset $R_s i_{dq}(t)$ of (1.14), $\bar{E}_c$ is generally not small compared to the available terminal voltage $v_{dq}(t)$ and is more complicated to treat. Thus, the model (1.17) is not used in this text.

Remark 1.1. Throughout this text,the linear map $\lambda_{dq}(t) = \mathbf{L}i_{dq} + \psi_{dq}$ is used for simplicity. If the flux is used as state, nonlinear effects, e.g. saturation, can be considered by simply replacing the linear map with a nonlinear one (ideally invertible to keep operations on sets simple). It is emphasized that no knowledge about the differential inductances [98] is required if the current is converted into flux externally, e.g. in (1.13) and (1.14). If the flux-current map is substituted into the state-space system, e.g. in (1.17), the complexity of treating nonlinear effects increases significantly.

1.4.2. Discrete Time

Control of modern drive systems is almost exclusively implemented on sampled digital control hardware. Thus, the continuous-time state-space model (1.14) is transformed into *discrete time*. Let $T_s \in \mathbb{R}_+$ be the sampling period and $k \in \mathbb{N}_+$ identify the discrete-time instant $t = kT_s$. For simplicity, it is assumed that $\omega(t)$ varies slowly with respect to the the sampling period.

Assumption 1.2. Let T_s be sufficiently small such that $\omega[k] \approx \omega(t)$ for all $t \in [kT_s, kT_s + T_s]$.

Moreover, the absolute value of $\omega[k]$ is limited according to

Assumption 1.3. Let $|\omega[k]|$ be smaller than the Nyquist frequency, i.e. $|\omega[k]| < \frac{\pi}{T_s}$.

This assumption is necessary to be able to compute or estimate $\omega[k]$ correctly from the state-space systems or from a position signal $\epsilon[k]$.

When performing discretization, a zero order hold element is typically assumed at the input of the continuous time system. In other words, a constant input is applied for the entire sampling period.

Assumption 1.4. Let $\bar{v}_{dq}[k] = \bar{v}_{dq}(t)$ for all $t \in [kT_s, kT_s + T_s)$.

First, the state-space model (1.13) is transformed into discrete time. Since the model defines an integrator dynamics, the discrete time dynamics can be obtained via integration

$$\lambda_{\alpha\beta}[k+1] = \lambda_{\alpha\beta}[k] + \bar{v}_{\alpha\beta}[k] \int_{kT_s}^{kT_s+T_s} d\tau = \lambda_{\alpha\beta}[k] + T_s \bar{v}_{\alpha\beta}[k], \tag{1.19}$$

where $\lambda_{\alpha\beta}[k+1] = \lambda_{\alpha\beta}(kT_s + T_s)$, $\lambda_{\alpha\beta}[k] = \lambda_{\alpha\beta}(kT_s)$, and $\bar{v}_{\alpha\beta}[k] = \bar{v}_{\alpha\beta}(kT_s)$, which is constant over the sampling instant. The state-space system (1.19) is linear time invariant (LTI). It defines an integrator dynamics with the poles at $1 \pm \mathrm{j}0$ in complex plane. Consequently, the undriven ($\bar{v}_{\alpha\beta}(t) \equiv 0$) autonomous system is marginally stable.

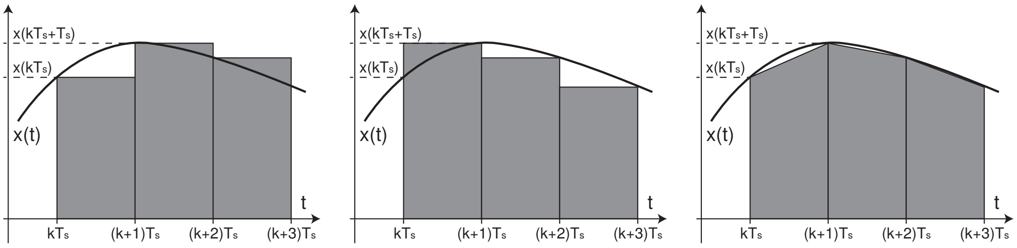
The continuous-time state-space model (1.14) can be transformed into discrete-time by exact (or zero-order-hold, ZOH) discretization [24], [28], [39], [58], which yields

$$\lambda_{dq}[k+1] = \mathbf{A}_d[k]\lambda_{dq}[k] + \mathbf{B}_d[k]\bar{v}_{dq}[k]. \tag{1.20}$$

with the parameter matrices (denoting $A_c = A_c(kT_s)$)

$$\mathbf{A}_d = \mathrm{e}^{\mathbf{A}_c T_s} = \begin{bmatrix} \cos(\omega[k]T_s) & \sin(\omega[k]T_s) \\ -\sin(\omega[k]T_s) & \cos(\omega[k]T_s) \end{bmatrix}; \tag{1.21a}$$

$$\begin{aligned} \mathbf{B}_d &= \left(\int_0^{T_s} \mathrm{e}^{\mathbf{A}_c \tau} d\tau \right) \mathbf{B}_c \qquad (1.21b) \\ &= \begin{cases} T_s \mathbf{B}_c = \begin{bmatrix} T_s & \\ & T_s \end{bmatrix} & \text{if } \omega[k] = 0, \\ \mathbf{A}_c^{-1}(\mathbf{A}_d - \mathbf{I})\mathbf{B}_c = \frac{1}{\omega[k]} \begin{bmatrix} \sin(\omega[k]T_s) & 1 - \cos(\omega[k]T_s) \\ \cos(\omega[k]T_s) - 1 & \sin(\omega[k]T_s) \end{bmatrix} & \text{otherwise.} \end{cases} \end{aligned}$$



(a) Euler forward (b) Euler backward (c) Tustin

Figure 1.6.: Approximate discretization

By Assumption 1.3, $\mathbf{B}_d$ is invertible, i.e.

$$\mathbf{B}_d^{-1} = \begin{cases} \frac{1}{T_s}\mathbf{I} & \text{if } \omega[k] = 0, \\ \frac{\omega[k]}{2}\begin{bmatrix} \frac{\sin(\omega[k]T_s)}{1-\cos(\omega[k]T_s)} & -1 \\ 1 & \frac{\sin(\omega[k]T_s)}{1-\cos(\omega[k]T_s)} \end{bmatrix} & \text{otherwise.} \end{cases}$$

By Assumption 1.2, the state-space system (1.20) with the parameter matrices (1.21) can be handled as LTI system, where $\mathbf{A}_d$, $\mathbf{B}_d$ are parametrized by the slow varying $\omega[k]$ (slow varying with respect to T_s). Some other useful properties of the state-space system are

(i) $\mathbf{A}_d$ is orthogonal ($\mathbf{A}_d^{-1} = \mathbf{A}_d'$ and $\mathbf{A}_d'\mathbf{A}_d = \mathbf{A}_d\mathbf{A}_d' = \mathbf{I}$)

(ii) $\mathbf{B}_d^{-1}(\mathbf{A}_d - \mathbf{I}) = -\omega[k]\mathbf{J} = \mathbf{A}_c(kT_s)$

(iii) $\mathbf{B}_d(\mathbf{B}_d')^{-1} = (\mathbf{B}_d')^{-1}\mathbf{B}_d = \mathbf{A}_d$ and $\mathbf{B}_d'(\mathbf{B}_d)^{-1} = (\mathbf{B}_d)^{-1}\mathbf{B}_d' = \mathbf{A}_d'$

Exact discretization preserves the stability properties of the continuous-time autonomous system, which is verified as follows

Proposition 1.2. *The undriven discrete-time system (1.20), using the parameter matrices (1.21) obtained via exact discretization, is marginally stable for all $\omega[k] \in \mathbb{R}$.*

Proof. Let $\Gamma[k] \overset{\text{def}}{=} \|\lambda_{dq}[k]\|$ be a candidate Lyapunov function ($\Gamma[k] > 0$ for all $\lambda_{dq}[k] \neq 0$; $\Gamma[k] = 0$ if $\lambda_{dq}[k] = 0$; $\Gamma[k]$ is radially unbounded and continuous in the origin). Since $\mathbf{A}_d$ is orthogonal, $\|\mathbf{A}_d\lambda_{dq}[k]\| = \|\lambda_{dq}[k]\|$ for all $\omega[k] \in \mathbb{R}$, which yields

$$\Gamma[k+1] - \Gamma[k] = \|\lambda_{dq}[k+1]\| - \|\lambda_{dq}[k]\| = \|\mathbf{A}_d\lambda_{dq}[k]\| - \|\lambda_{dq}[k]\| = 0. \qquad (1.22)$$

Since $\Gamma[k+1] - \Gamma[k]$ is nonpositive, the system is (marginally) stable. □

Exact discretization is sometimes intractable or the resulting discrete-time system is difficult to handle in practice. Thus, approximations are commonly used to simplify the discretization process as shown in Figure 1.6. The most popular approximations are the forward Euler, the backward Euler, and the Tustin method, each with advantages and drawbacks [24], [28], [39], [58].

These approximations can be applied to drive systems assuming that $\omega[k]T_s$ is sufficiently small, i.e. $|\omega[k]| \ll \frac{\pi}{T_s}$. Using the Tustin method, the parameter matrices are approximated without case distinctions (denoting $A_c = A_c(kT_s)$)

$$\begin{aligned}\mathbf{A}_d &\approx \left(\mathbf{I} + \frac{1}{2}\mathbf{A}_c T_s\right)\left(\mathbf{I} - \frac{1}{2}\mathbf{A}_c T_s\right)^{-1} \qquad (1.23a)\\ &= \frac{1}{4 + (\omega[k]T_s)^2}\begin{bmatrix} 4 - (\omega[k]T_s)^2 & 4(\omega[k]T_s)^2 \\ -4(\omega[k]T_s)^2 & 4 - (\omega[k]T_s)^2 \end{bmatrix};\\ \mathbf{B}_d &\approx \begin{cases} T_s\mathbf{B}_c & \text{if } \omega[k] = 0, \\ \mathbf{A}_c^{-1}(\mathbf{A} - \mathbf{I})\mathbf{B}_c & \text{otherwise.} \end{cases} \qquad (1.23b)\\ &= \frac{1}{4 + (\omega[k]T_s)^2}\begin{bmatrix} 4T_s & 2\omega[k]T_s^2 \\ -2\omega[k]T_s^2 & 4T_s \end{bmatrix}.\end{aligned}$$

The Tustin method preserves the properties (i), (ii), and (iii) of the parameter matrices obtained via exact discretization. Moreover, the Tustin approximation is known to preserve also the stability properties of the continuous time system, which is verified as follows

Proposition 1.3. *The undriven discrete-time system (1.20), using the parameter matrices (1.23) obtained via Tustin discretization, is marginally stable for all $\omega[k] \in \mathbb{R}$.*

Proof. $\mathbf{A}_d$ obtained via Tusin approximation is orthogonal. Thus, the proof is similar to the proof of Proposition 1.2. □

In contrast, the Euler forward and Euler backward do not preserve the properties of the continuous time system, i.e. the discrete time system obtained via exact discretization. For instance, the Euler forward method yields the parameter matrices

$$\begin{aligned}\mathbf{A}_{euler} &= (\mathbf{I} + \mathbf{A}_c T_s) = \begin{bmatrix} 1 & \omega[k]T_s \\ -\omega[k]T_s & 1 \end{bmatrix};\\ \mathbf{B}_{euler} &= \begin{cases} T_s\mathbf{B}_c & \text{if } \omega[k] = 0, \\ \mathbf{A}_c^{-1}(\mathbf{A} - \mathbf{I})\mathbf{B}_c & \text{otherwise.} \end{cases} = \begin{bmatrix} T_s & \\ & T_s \end{bmatrix}.\end{aligned}$$

The matrix $\mathbf{A}_{euler}$ is clearly not orthogonal but $\mathbf{B}_{euler}^{-1}(\mathbf{A}_{euler} - \mathbf{I}) = -\omega[k]\mathbf{J} = \mathbf{A}_c(kT_s)$. Moreover, the Euler forward method does not preserve the stability properties of the continuous time system, which is highlighted as follows

Proposition. *The undriven discrete-time system (1.20), using the parameter matrix* $\mathbf{A}_{euler}$ *obtained via Euler approximation, is marginally stable iff* $\omega[k] = 0$ *and unstable otherwise.*

Proof. Let $\Gamma[k] \overset{\text{def}}{=} \|\lambda_{dq}[k]\|$ be a candidate Lyapunov function ($\Gamma[k] > 0$ for all $\lambda_{dq}[k] \neq 0$; $\Gamma[k] = 0$ if $\lambda_{dq}[k] = 0$; $\Gamma[k]$ is radially unbounded and continuous in the origin), which yields

$$\begin{aligned}\Gamma[k+1] - \Gamma[k] &= \|\lambda_{dq}[k+1]\| - \|\lambda_{dq}[k]\| = \|\mathbf{A}_d[k]\lambda_{dq}[k]\| - \|\lambda_{dq}[k]\| \\ &= \lambda'_{dq}[k]\mathbf{A}'[k]\mathbf{A}[k]\lambda_{dq}[k] - \lambda'_{dq}[k]\lambda_{dq}[k] \\ &= (1 + \omega^2[k]T_s^2)(\lambda_d^2[k] + \lambda_q^2[k]) - (\lambda_d^2[k] + \lambda_q^2[k]) \\ &= \omega^2[k]T_s^2(\lambda_d^2[k] + \lambda_q^2[k]). \qquad (1.25)\end{aligned}$$

Thus, $\Gamma[k+1] - \Gamma[k]$ is nonpositive (stable) iff $\omega[k] = 0$. Otherwise, the system is unstable. □

Since the properties (i), (ii), and (iii) of the parameter matrices and the stability properties of the undriven system are useful in control design, the discrete time system, which is obtained via exact discretization or Tustin method (without distinction), is used and the Euler forward and backward approximation is ignored throughout this text.

1.5. Steady-State Constraints

An electric machine cannot be operated with arbitrarily large voltages and currents due to the finite terminal voltage supplied by the inverter and thermal constraints. In this section, the constraints are shown, which have to be satisfied (at least) in steady-state conditions. The constraints are defined in the current or flux space, and can be transformed between these spaces using the linear transformation (1.10). Constraints, which are shown using quantities without time (t) or sampling instant $[k]$ are valid in both continuous-time and discrete-time.

The magnitude of the stator current needs to be limited since a winding is able to transmit only a finite amount of heat, i.e. power losses, to the ambient without exceeding its maximum operating temperature. Since the winding losses increase with the magnitude of the current, the **current constraint**

$$i_{dq} \in \mathcal{I} = \left\{ i_{dq} \in \mathbb{R}^2 \mid \|i_{dq}\| \leq I_r \right\}, \tag{1.26}$$

is introduced, where $I_r \in \mathbb{R}_+$ is the rated current and the time dependency (t) is omitted for compactness. However, the thermal time constants are orders of magnitude higher than the electric ones. Thus, the current constraint needs to be satisfied only in steady-state conditions. Transient violations of the constraint are accepted.

An inverter provides a terminal voltage with a finite maximum magnitude. For analysis, the largest time invariant (with respect to the dq transformation angle ϵ) set is used. The set depends on the DC link voltage v_c and is defined as

$$v_{dq} \in \mathcal{V} = \left\{ v_{dq} \in \mathbb{R}^2 \mid \|v_{dq}\| \leq v_r \overset{\text{def}}{=} \frac{v_c}{\sqrt{3}} \right\}, \tag{1.27}$$

where $v_r \in \mathbb{R}_+$ is the rated voltage and $v_c \in \mathbb{R}_+$ is the DC link voltage. Details on the terminal voltage set are treated in Chapter 3.

As shown in the previous section, the treatment can be simplified introducing the compensated voltage as input

$$\bar{v}_{dq} = v_{dq} - R_s i_{dq}. \tag{1.28}$$

Using the compensated terminal voltage as input, it must be ensured that $\bar{v}_{dq}$ can be applied to the system. In other words, the existence of a terminal voltage $v_{dq} \in \mathcal{V}$ for all $i_{dq} \in \mathcal{I}$ is required. Thus, the compensated terminal voltage has to satisfy the **voltage constraint**

$$\bar{v}_{dq} \in \mathcal{V} \ominus R_s \mathcal{I} = \left\{ \bar{v}_{dq} \in \mathbb{R}^2 \mid \|\bar{v}_{dq}\| \leq v_r - R_s I_r \right\}, \tag{1.29}$$

where $\ominus$ denotes the Pontryagin difference. Clearly, some of the available terminal voltage is lost above all if $\|i_{dq}\| \ll I_r$. However, this approach is justified since the resistive voltage drop is typically small compared to the DC link voltage, i.e. $R_s \|i_{dq}\| \ll v_r$.

The compensated voltage approach can be generalized to take model uncertainties (see Section 2.5) and nonlinear inverter behavior (see Section 3.2) into account. The safety factor $\rho_v \in (0, 1)$ is introduced and defines the rated compensated terminal voltage set as

$$\bar{v}_{dq} \in \bar{\mathcal{V}} \overset{\text{def}}{=} \left\{ \bar{v}_{dq} \in \mathbb{R}^2 \mid \|\bar{v}_{dq}\| \leq \bar{v}_r \overset{\text{def}}{=} \rho_v v_r \right\}, \tag{1.30}$$

The factor ρ_v is usually chosen heuristically such that $\rho_v v_r \leq v_r - R_s I_r$. Typical values are $\rho_v = (0.8, 1)$.

In the constrained system, steady-state $\dot{\lambda}_{dq}(t) = \mathbf{A}_c \lambda_{dq}(t) + \mathbf{B}_c \bar{v}_{dq}(t) = 0$ can be achieved if and only if (iff)

$$-\mathbf{B}_c^{-1}\mathbf{A}_c \lambda_{dq}(t) = \omega(t)\mathbf{J}\lambda_{dq}(t) \in \bar{\mathcal{V}}. \tag{1.31}$$

The same result is obtained using the discrete-time state-space system (exact, Tustin, or Euler method), where steady-state $\lambda_{dq}[k+1] = \lambda_{dq}[k] = \mathbf{A}\lambda_{dq}[k] + \mathbf{B}\bar{v}_{dq}[k]$ is obtained iff

$$-\mathbf{B}^{-1}(\mathbf{A} - \mathbf{I})\lambda_{dq}[k] = \omega[k]\mathbf{J}\lambda_{dq}[k] \in \bar{\mathcal{V}}. \tag{1.32}$$

Since $\mathbf{J}^{-1} \circ \bar{\mathcal{V}} = \bar{\mathcal{V}}$, where $\circ$ denotes the convolution, the following steady-state constraint is obtained

$$\omega\lambda_{dq} \in \bar{\mathcal{V}}, \tag{1.33}$$

Equivalently, the condition can be written using the ω contracted set $\bar{\mathcal{V}}$, i.e. Λ, with

$$\lambda_{dq} \in \Lambda = \begin{cases} \mathbb{R}^2 & \text{if } \omega = 0, \\ \frac{1}{\omega}\bar{\mathcal{V}} & \text{otherwise.} \end{cases} = \left\{\lambda_{dq} \in \mathbb{R}^2 \mid |\omega| \|\lambda_{dq}\| \leq \bar{v}_r\right\}. \tag{1.34}$$

Combining the current and flux constraints yields

$$\begin{aligned} i_{dq} \in \mathcal{I} \ \cap \ &\left(\bar{\Lambda} \overset{\text{def}}{=} \mathbf{L}^{-1} \circ \Lambda - \mathbf{L}^{-1}\psi_{dq}\right) \\ &= \left\{i_{dq} \in \mathbb{R}^2 \mid \|i_{dq}\| \leq I_r \text{ and } |\omega| \|\mathbf{L}i_{dq} + \psi_{dq}\| \leq \bar{v}_r\right\} \end{aligned} \tag{1.35}$$

The sets and their intersection are shown in Figure 2.1(a) in Chapter 2. Equivalently, the conditions are satisfied if

$$\begin{aligned} \lambda_{dq} \in \ &\left(\bar{\mathcal{I}} \overset{\text{def}}{=} \mathbf{L} \circ \mathcal{I} + \psi_{dq}\right) \ \cap \ \Lambda \\ &= \left\{\lambda_{dq} \in \mathbb{R}^2 \mid \|\mathbf{L}^{-1}(\lambda_{dq} - \phi_{dq})\| \leq I_r \text{ and } |\omega| \|\lambda_{dq}\| \leq \bar{v}_r\right\} \end{aligned} \tag{1.36}$$

These sets and their intersection are portrayed in Figure 2.1(b) in Chapter 2.

The sets (1.35) and (1.36) define the high speed behavior of a PMSM. The location of the current and voltage constraint with respect to each other defines whether a machine has an electrically limited maximum speed or not [10]. The center of $\bar{\Lambda}$ is $i_c \overset{\text{def}}{=} [-\psi/Ld, 0]'$. If $i_c \notin \mathcal{I}$, the PMSM has a limited maximum speed

ω_m since for $|\omega| \to \infty$ there exists no current satisfying (1.35). Contrarily, a PMSM with $i_c \in \mathcal{I}$ has the maximum speed $\omega_m = \infty$ since there always exists a current (e.g. i_c), which satisfies (1.35). Thus, a PMSM is subject to the **speed constraint**

$$\omega \leq \omega_m \overset{\text{def}}{=} \begin{cases} \frac{\bar{v}_r}{\psi - L_d I_r} & \text{if } \frac{\psi}{L_d} > I_r, \\ \infty & \text{otherwise.} \end{cases} \tag{1.37}$$

Examples are shown in Chapter 2, where a machine with $\omega_m = \infty$ is depicted in Figure 2.2 and a machine with $\omega_m < \infty$ is displayed in Figure 2.3.

Chapter 2.

Electromagnetic Torque

In this chapter, the torque generation of the PMSM is explained. The maximum (absolute value) torque of a PMSM depends on the operating condition specifically the DC link voltage and the machine speed. The maximum torque problem is defined and a way to compute it efficiently is shown. Moreover, the current, i.e. flux, which produces a desired amount of torque is not unique. A typical approach for choosing the currents is to maximize the machine efficiency. The related optimization problem is defined and a way to compute it efficiently is shown. The chapter is concluded addressing typical PMSM model inconsistency. The real world PMSM is nonlinear due to saturation and the PMSM model based on the linear current-flux relationship is a local approximation at best. Thus, some approaches to improve the model correspondence with the real system are shown.

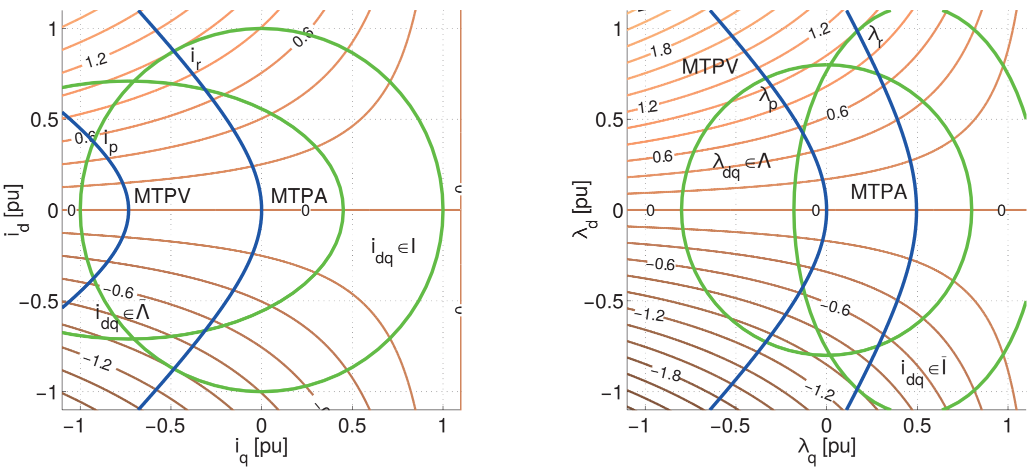
2.1. Torque Equation

In an electrical machine, torque is produced by driving currents in the armature winding. The current interacts with the magnetic field, i.e. flux, and produces the electromagnetic torque $T \in \mathbb{R}$. The torque equation is obtained as power balance of the electromechanical energy conversion neglecting the resistive voltage drop [5], [78], [100]

$$\frac{3}{2} v_{dq}' i_{dq} = \frac{3}{2} \left(v_d i_d + v_q i_q \right) = T \omega_m + \frac{dW_m}{dt}, \tag{2.1}$$

where the mechanical rotor speed is ω_m and the magnetic energy is W_m. Substituting $d\omega_m / \epsilon_m$ for dt and rearranging the equation yields [5], [100]

$$T = \frac{3}{2} p \; i_{dq}' \mathbf{J} \lambda_{dq} + \frac{\partial W_{cm}}{\partial \epsilon_m} = \frac{3}{2} p \; \left(\lambda_d i_q - \lambda_q i_d \right) + \frac{\partial W_{cm}}{\partial \epsilon_m}, \tag{2.2}$$



(a) Current space

(b) Flux space

Figure 2.1.: State-space with isotorque loci; the sets $i_{dq} \in \mathcal{I}$ and $\lambda_{dq} \in \Lambda$; MTPA and MTPV trajectory

where the parameter $p \in \mathbb{N}_+$ is the number of pole pairs of the electrical machine. The second term $\partial W_{cm}/\partial \epsilon_m$ is the derivation of the magnetic coenergy with respect to the rotor position, which has no average value in steady-state conditions. Thus, it is typically omitted when studying PMSM control strategies [5].

The torque equation of a three-phase armature winding (2.2) can specified for the PMSM assuming linear flux dependencies. Substituting the flux equations (1.10), the torque equation is written as function of the currents

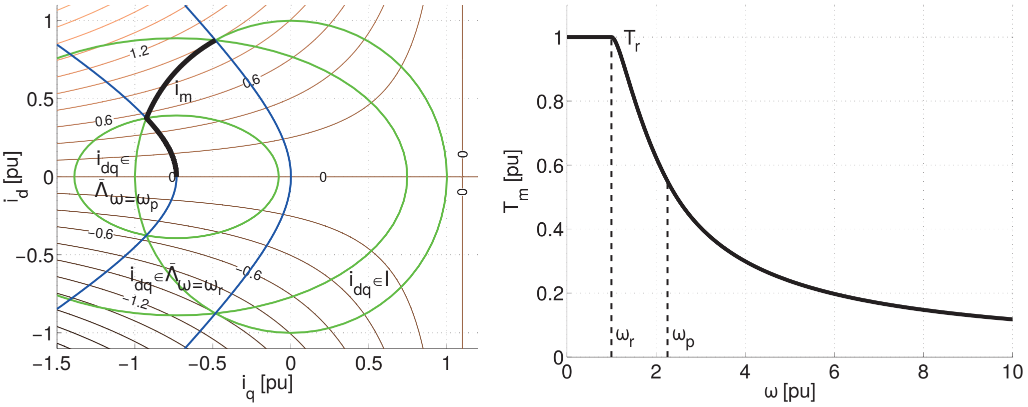
$$T = \frac{3}{2}p\,\left(i'_{dq}\mathbf{J}\mathbf{L}i_{dq} + i'_{dq}\mathbf{J}\psi_{dq}\right) = \frac{3}{2}p\,\left(\psi + (L_d - L_q)i_d\right)i_q. \tag{2.3}$$

The electromagnetic torque T modes have two components. The PM torque $3/2p\psi i_q$ is produced by interaction of the rotor flux generated by the PM and the stator current. The reluctance torque $3/2p(L_d - L_q)i_d i_q$ is produced by the tendency of the flux to follow a minimum reluctance path.

Equivalently, the torque equation (2.2) can be written as function of fluxes substituting the inverse of (1.10). In this case the torque equation becomes

$$T = \frac{3}{2}p\,\left(\lambda'_{dq}\mathbf{L}^{-1}\mathbf{J}\lambda_{dq} - \psi'_{dq}\mathbf{L}^{-1}\mathbf{J}\lambda_{dq}\right) = \frac{3}{2}p\left(\frac{\psi}{L_d} + \left(\frac{1}{L_q} - \frac{1}{L_d}\right)\lambda_d\right)\lambda_q. \tag{2.4}$$

The PMSM torque equations (2.3) and (2.4) are studied in the next sections for $L_d \leq L_q$, which is satisfied by all surface mounted and interior PMSMs [10]. When



(a) Current space

(b) Maximum torque vs. speed

Figure 2.2.: Maximum torque characteristic of a machine with $\omega_m = \infty$

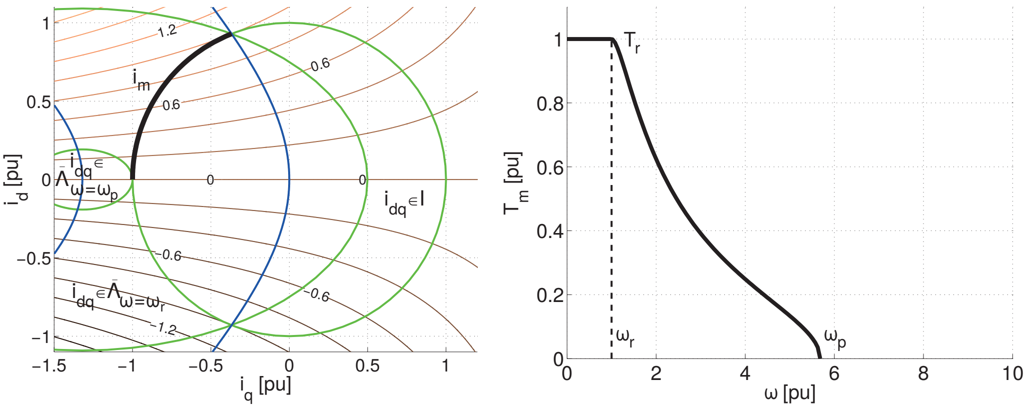
studying the maximum torque dependencies and optimal operation conditions, it is convenient to consider symmetry properties of the torque equation, which can be observed in Figure 2.1. The current limit (1.35) is symmetric with respect to the d axis, i.e. $i_q = 0$. Similarly, the voltage constraint (1.36) is symmetric with respect to the d axis due to the introduction of the compensated voltage (1.28). Moreover, the torque equation (2.3), i.e. (2.4), has the following property

$$T\left([i_d, i_q]'\right) = -T\left([i_d, -i_q]'\right); \quad T\left([\lambda_d, \lambda_q]'\right) = -T\left([\lambda_d, -\lambda_q]'\right). \tag{2.5}$$

Thus, the maximum torque dependencies and optimal operation conditions are studied for positive torque values only. The results cover negative torque values as well by changing the sign of i_q, i.e. λ_q.

2.2. Maximum Torque

The current, i.e. flux, is bounded by (1.35), i.e. (1.36) in steady-state conditions. Thus, the maximum (and minimum) available torque is limited as well. In this section, an approach to find the maximum (steady-state) torque $T_m \in \mathbb{R}_+$, which can be provided by the PMSM, is shown. Due to the symmetries of the torque equation, finding T_m is equivalent to finding the minimum torque, which is the additive inverse of T_m, i.e. $-T_m$.



(a) Current space

(b) Maximum torque vs. speed

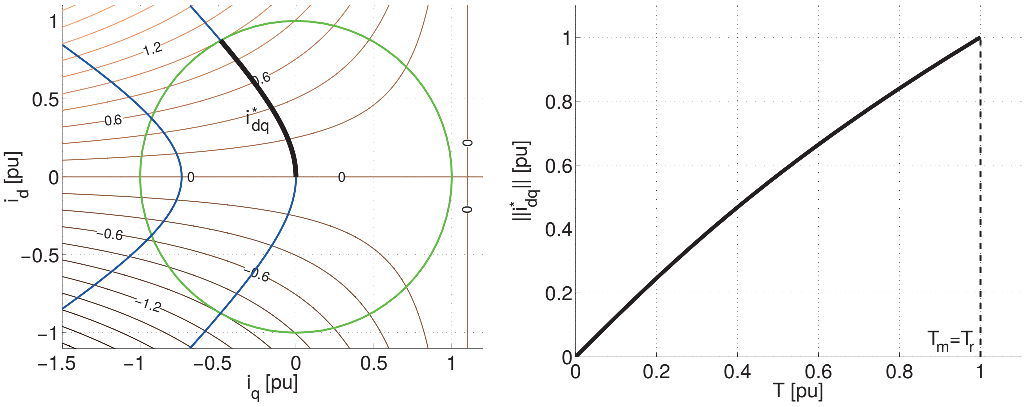
Figure 2.3.: Maximum torque characteristic of a machine with $\omega_m < \infty$

The maximum torque is defined to be

$$\begin{aligned} T_m \overset{\text{def}}{=} \frac{3}{2}p \qquad & \max_{i_{dq},\lambda_{dq}} \quad i'_{dq}\mathbf{J}\lambda_{dq} && \text{(2.6a)} \\ \text{subject to} \quad & \|i_{dq}\| \leq I_r; && \text{(2.6b)} \\ & |\omega|\|\lambda_{dq}\| \leq \bar{v}_r; && \text{(2.6c)} \\ & \lambda_{dq} = \mathbf{L}i_{dq} + \psi_{dq} && \text{(2.6d)} \end{aligned}$$

The problem uses the torque equation (2.6a) as objective function. The steady-state current (2.6b) and voltage (2.6c) limit define the set where the current, i.e. fluxes, can lie. Due to the $|\omega|$ dependence of the voltage limit (2.6c), the maximum torque T_m depends on the machine speed. Moreover, the linear equality constraint (2.6d) defines the relation of currents and fluxes. The problem (2.6) can be written as Quadratic Constrained Quadratic Program (QCQP) in standard form. However, the resulting matrices are indefinite and the problem is NP hard to solve in general [20]. Moreover, it can be infeasible dependent on the parameters.

Thus, T_m is rarely computed directly but using system properties, e.g. $L_d \leq L_q$, and exploiting the low dimensionality of the problem [10], [80]–[82]. Particularly helpful are the Maximum Torque per Ampere (MTPA) and Maximum Torque per Volt (MTPV) trajectories, which are shown in Figure 2.1. These trajectories and the intersections with the current and flux loci are derived in Appendix B.



(a) Current space

(b) Current magnitude vs. torque

Figure 2.4.: Optimal operation in base mode (constant torque mode)

2.2.1. Base Mode

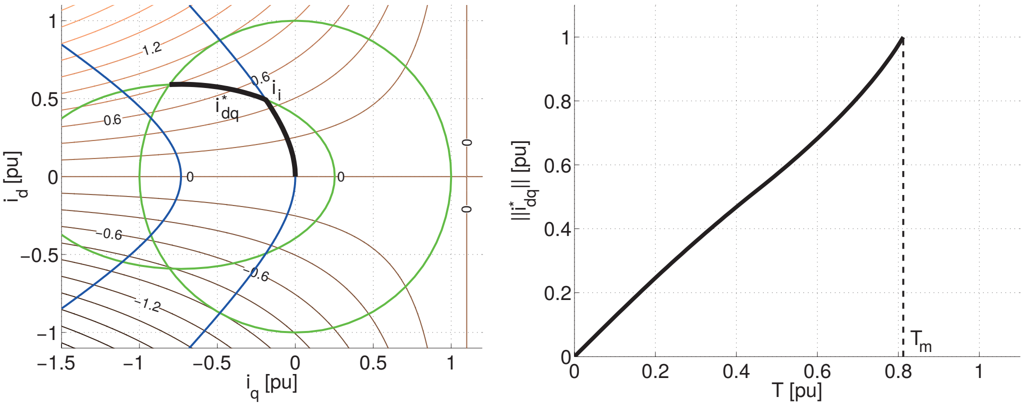
At low velocities ($\omega \approx 0$), the voltage constraint (2.6c) is always true. The torque T_m is produced by the current $i_m \in \mathbb{R}^2$, which lies on the MTPA trajectory and is denoted as $i_m \in \text{MTPA}$. Moreover, i_m lies on the largest admissible isocurrent locus, i.e. the border of $\mathcal{I}$, which is denoted as $i_m \in \operatorname{br}\mathcal{I}$. Thus, $i_m \in \text{MTPA} \cap \operatorname{br}\mathcal{I}$. This intersection defines a set containing two current vectors, which produces the maximum and minimum torque.

The solution $i_q > 0$ is named *rated* operation point [10], [80]–[82], where the machine uses the rated current $i_r \in \mathbb{R}^2$ and the rated flux $\lambda_r \overset{\text{def}}{=} \mathbf{L}i_r + \psi_{dq}$ to produce the rated torque $T_r \overset{\text{def}}{=} 3/2p\ i_r'\mathbf{J}\lambda_r$. These rated values are parameters of a PMSM and are shown in Figure 2.1.

The PMSM is said to work in *base mode* or *constant torque mode* when the maximum torque T_m is constant and corresponds to the rated torque T_r [10], [80]–[82]. The machine can produce the rated torque T_r iff $\lambda_r \in \Lambda$. Since Λ shrinks increasing $|\omega|$, the machine works in constant torque mode if

$$|\omega| \leq \omega_r \overset{\text{def}}{=} \frac{\bar{v}_r}{\|\lambda_r\|}. \tag{2.7}$$

Consequently, the rated speed ω_r and the rated power $P_r = T_r\omega_r$ of a drive system depend on the rated voltage v_r, i.e. the DC link voltage v_c.



(a) Current space

(b) Current magnitude vs. torque

Figure 2.5.: Optimal operation in field weakening (constant power mode)

2.2.2. Field Weakening

The PMSM is said to work in *field weakening*, if the speed is higher than the rated one ($|\omega| > \omega_r$) [18], [103]. In this condition, the rated flux $\lambda_r \notin \Lambda$ and the rated torque T_r cannot be achieved. Thus, the updated maximum torque $T_m < T_r$ must be found.

Considering the positive halfplane ($i_q \geq 0$, $\lambda_q \geq 0$), the torque increases along the largest isoflux locus $\operatorname{br}\Lambda$ starting at the d axis until its peak is achieved when intersecting the MTPV trajectory. Consequently, T_m is obtained at the intersection $\mathrm{MTPV} \cap \operatorname{br}\Lambda$ if this point satisfies $i_{dq} \in \mathcal{I}$. Otherwise, T_m is obtained at the intersection $i_{dq} \in \operatorname{br}\mathcal{I}$ with $\lambda_{dq} \in \operatorname{br}\Lambda$.

To simplify the treatment, the *rated power* operation point is introduced, which defines the current i_p and the flux $\lambda_p \overset{\text{def}}{=} \mathbf{L}i_p + \psi_{dq}$. It is defined to be the intersection of the MTPV trajectory with the isocurrent locus $\operatorname{br}\mathcal{I}$, if it exists. This case is shown in Figure 2.1 and Figure 2.2. If $\mathrm{MTPV} \cap \operatorname{br}\mathcal{I} = \emptyset$, the rated power operation point is defined to be $i_p = [-I_r, 0]'$. This case is shown in Figure 2.3.

If $\lambda_p \in \Lambda$ (and $\lambda_r \notin \Lambda$), the machine is said to work in *constant (apparent) power mode* [10], [80]–[82]. In this mode, the maximum torque T_m is obtained at the intersection of $i_{dq} \in \operatorname{br}\mathcal{I}$ with $\lambda_{dq} \in \operatorname{br}\Lambda$. Similar to the constant torque mode, the machine works in constant power mode if

$$\omega_r < |\omega| \leq \omega_p \overset{\text{def}}{=} \frac{\bar{v}_r}{\|\lambda_p\|}. \tag{2.8}$$

Machines with electrically limited maximum speed $\omega_m < \infty$ cannot exceed the constant power mode, since $\omega_p \equiv \omega_m$. On the other hand, machines with $\omega_m = \infty$ can achieve $|\omega| > \omega_p$, which is named *reduced (apparent) power mode* [10], [80]–[82]. In reduced power mode, the intersection $\text{MTPV} \cap \operatorname{br} \Lambda$ satisfies $i_{dq} \in \mathcal{I}$ and defines therefore the current, i.e. flux, which provides T_m.

The maximum torque T_m provided by a machine with limited maximum speed $\omega_m < \infty$ is shown in Figure 2.3. The maximum torque T_m provided by a machine with infinite maximum speed $\omega_m = \infty$ is shown in Figure 2.2.

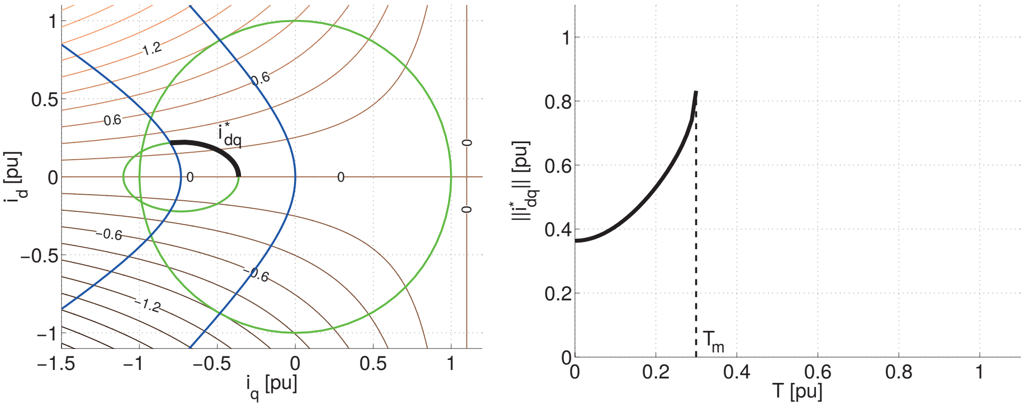
2.3. Optimal Operation

An electrical machine is designed to apply the electromagnetic torque T to the drive shaft. The current, i.e. flux, which produces T is not unique and the degree of freedom is exploited to increase the drive system efficiency. This is typically formulated as minimization problem of the electric losses [10], [80]–[82], which increase with the current magnitude. Thus, the operation is said to be *optimal*, if the torque T is generated by the current $i^\star_{dq}$ and flux $\lambda^\star_{dq}$, which solve

$$\begin{aligned}
\underset{i_{dq},\lambda_{dq}}{\text{minimize}} \quad & \|i_{dq}\| && \text{(2.9a)}\\
\text{subject to} \quad & \|i_{dq}\| \le I_r; && \text{(2.9b)}\\
& |\omega|\|\lambda_{dq}\| \le \bar{v}_r; && \text{(2.9c)}\\
& \lambda_{dq} = \mathbf{L}i_{dq} + \psi_{dq}; && \text{(2.9d)}\\
& 3/2p\ i'_{dq}\mathbf{J}\lambda_{dq} = T; && \text{(2.9e)}\\
& |T| \le T_m && \text{(2.9f)}
\end{aligned}$$

The problem uses the current magnitude (2.9a) as objective function. The current (2.9b) and voltage (2.9c) constraint needs to be considered when solving for the optimal states and the relation between currents and fluxes is defined by (2.9d). The constraint (2.9e) defines the required torque T, which must satisfy the (speed-dependent) maximum torque T_m constraint (2.9f). Clearly, solving (2.9) directly leads to a similar problem set than solving (2.6). Thus, (2.9) is solved using system properties, e.g. $L_d \le L_q$, and the low dimensionality of the problem [80]–[82].

In base mode ($|\omega| \le \omega_r$), the voltage limit can be safely neglected. Thus, the optimal current $i^\star_{dq}$ and flux $\lambda^\star_{dq}$ lie on the MTPA trajectory by definition of the



(a) Current space

(b) Current magnitude vs. torque

Figure 2.6.: Optimal operation in field weakening (reduced power mode)

trajectory. The currents $i^\star_{dq}$ are shown in Figure 2.4(a) and the (minimum) current magnitude $\|i^\star_{dq}\|$, which is necessary to produce the torque T is shown in Figure 2.4(b).

In field weakening ($|\omega| > \omega_r$), the voltage limit needs to be taken into account. The optimal states depend on whether there exists an intersection between the MTPA and the largest isoflux trajectory $\lambda_{dq} \in \operatorname{br} \Lambda$ at the flux λ_i, i.e. i_i (Figure 2.5(a)), or not, i.e. $\text{MTPA} \cap \operatorname{br} \Lambda = \emptyset$ (Figure 2.6(a)). Clearly, there exists an intersection if Λ contains the center of $i_{dq} \in \mathcal{I}$ in the flux space, i.e. $\psi \in \Lambda$, which is equivalent to the condition

$$|\omega|\psi \leq \bar{v}_r. \tag{2.10}$$

If there exists an intersection, $\lambda^\star_{dq} \in \text{MTPA}$ for low torque values. For high torque values, $\lambda^\star_{dq} \in \operatorname{br} \Lambda$. This case is shown in Figure 2.5(a) and the (minimum) current magnitude $\|i^\star_{dq}\|$, which is necessary to produce the torque T is shown in Figure 2.5(b). Increasing further $|\omega|$, $\text{MTPA} \cap \operatorname{br} \Lambda = \emptyset$ and $\lambda^\star_{dq} \in \operatorname{br} \Lambda$. This case is shown in Figure 2.6(a) and the (minimum) current magnitude $\|i^\star_{dq}\|$, which is necessary to produce the torque T is shown in Figure 2.6(b).

Figure 2.5 and Figure 2.6 show operation in the constant power mode and reduced power mode, respectively. However, whether $\text{MTPA} \cap \operatorname{br} \Lambda = \emptyset$ or an intersection exists, is not related to the operation mode.

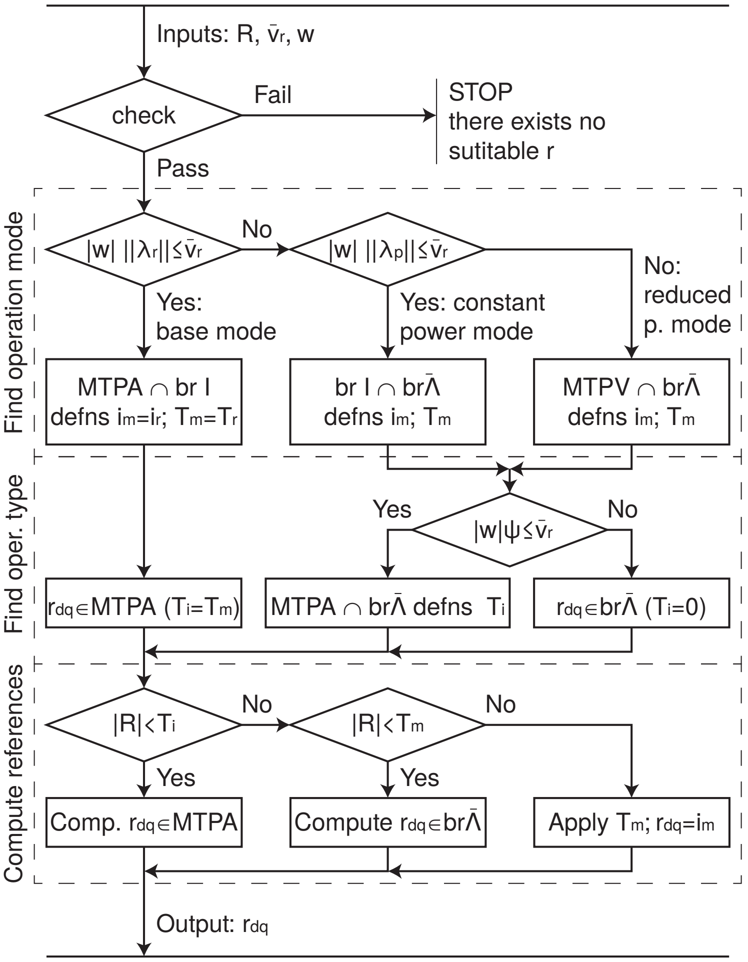
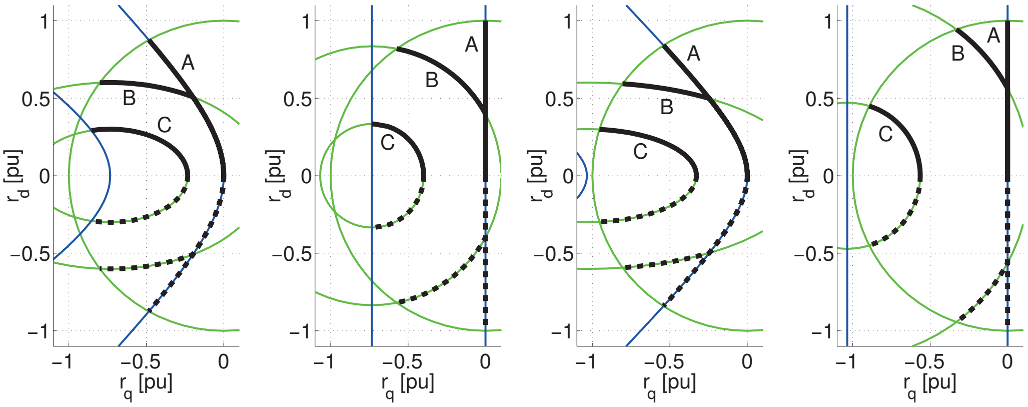


Figure 2.7.: Reference r_{dq} generation: flux diagram

2.4. Reference Generation Procedure

Based on the torque reference R and the measurements v_c and ω, a current reference vector r_{dq} is generated. This vector serves as reference input of a control algorithm. The vector r_{dq} defines the desired steady-state operating condition and is required to satisfy the constraints and optimal operating conditions. From r_{dq}, the dq flux reference vector $\bar{r}_{dq}$ is obtained by $\bar{r}_{dq} = \mathbf{L}r + \psi_{dq}$. Similarly, the $\alpha\beta$ flux reference vector $\bar{r}_{\alpha\beta}$ is obtained via inverse Park transformation.

The reference generation is carried out in four steps and it is described by the block diagram in Figure 2.7. The first step is to check the measurements to ensure that there



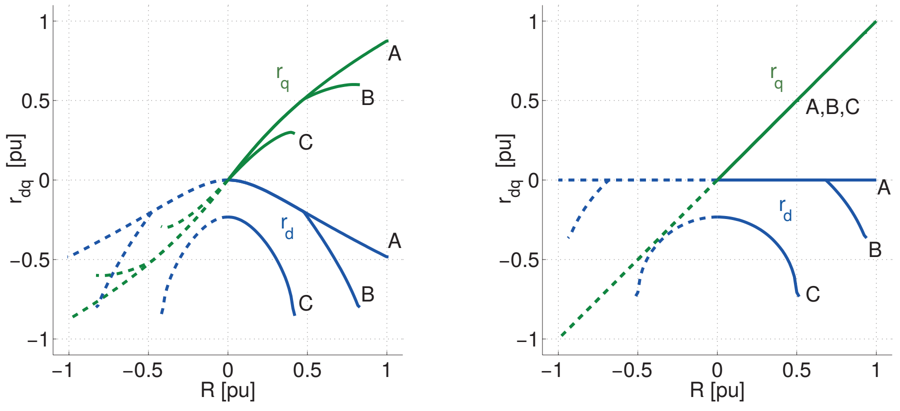
(a) $L_d < L_q$; $\omega_m = \infty$ (b) $L_d = L_q$; $\omega_m = \infty$ (c) $L_d < L_q$; $\omega_m < \infty$ (d) $L_d = L_q$; $\omega_m < \infty$

Figure 2.8.: Reference current r_{dq} for $R \in [0, \infty)$ (solid) and $R \in (-\infty, 0]$ (dashed) in the current plane

exists a suitable reference r_{dq}. The DC link voltage is required to be positive $\bar{v}_r \geq 0$ since $\Lambda = \emptyset$ otherwise and no $r_{dq} \in \bar{\Lambda}$ exists. Moreover, $|\omega|$ is required to not exceed the maximum velocity ω_m. This check is only necessary for PMSM with $\omega_m < \infty$ where the set $0 \notin \bar{\mathcal{I}}$, i.e. $\mathbf{L}^{-1}\psi_{dq} = [\psi/L_d, 0]' \notin \mathcal{I}$ ($\psi/L_d > I_r$). If such a machine exceeds its rated power region $|\omega|\|\lambda_p\| > \bar{v}_r$ or equivalently $|\omega|(\psi - L_d I_r) > \bar{v}_r$, the drive system must be stopped since $\mathcal{I} \cap \bar{\Lambda} = \emptyset$. In other words, there exists no r_{dq}, which satisfies (1.35) in these conditions. If these conditions fail, no r_{dq} can be computed and the drive system must be stopped.

The second step is identifying the operation mode (constant torque mode, constant power mode, reduced power mode) and computing the maximum torque T_m. The rated operation point λ_r, i_r, which defines the rated torque T_r are parameters of the system. They can be computed offline with $i_r \in \text{MTPA} \cap \operatorname{br} \mathcal{I}$ (with $i_q > 0$). If $|\omega|\|\lambda_r\| \leq \bar{v}_r$ the drive system works in base mode (constant torque mode) and the maximum torque is $T_m = T_r$ and is generated by $i_m = i_r$. Otherwise, the drive system works in field weakening.

Similar to the rated operation point, the operation point λ_p, which identifies operation in constant power mode, is a parameter of the system and can be obtained offline. For machines with $\omega_m = \infty$, it is obtained by $\lambda_p \in \text{MTPV} \cap \bar{\mathcal{I}}$. For machines with $\omega_m < \infty$, it is $\lambda_p = [\psi - L_d I_r, 0]'$. Thus, if $|\omega|\|\lambda_p\| \leq \bar{v}_r < |\omega|\|\lambda_r\|$, the drive system works in field weakening (constant power mode) and the maximum torque T_m is obtained at $i_m \in \operatorname{br} \bar{\Lambda} \cap \operatorname{br} \mathcal{I}$ (with $i_q > 0$). If $|\omega|\|\lambda_p\| > \bar{v}_r$, the drive system works in field weakening (reduced power mode) and the maximum torque T_m is obtained



(a) $L_d < L_q$; $\omega_m = \infty$

(b) $L_d = L_q$; $\omega_m = \infty$

Figure 2.9.: Reference current r_{dq} as function of $R \in [0, T_m]$ (solid) and $R \in [-T_m, 0]$ (dashed)

at $i_m \in \mathrm{MTPV} \cap \mathrm{br}\,\bar{\Lambda}$ (with $i_q > 0$).

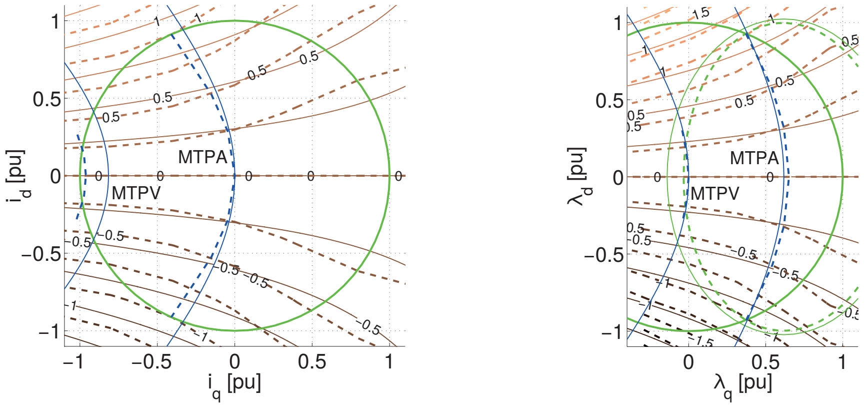
The third step is identifying the operation type, i.e. operation on the MTPA trajectory, on the largest isoflux locus $\mathrm{br}\,\bar{\Lambda}$, or at maximum torque. Therefore, the intersection torque T_i is defined. If $|R| < T_i$, the drive system is operated with $r_{dq} \in \mathrm{MTPA}$. If $T_i \leq |R| < T_m$, the drive system is operated with $r \in \mathrm{br}\,\bar{\Lambda}$. Otherwise, the torque reference magnitude $|R| \geq T_m$ corresponds or exceeds the maximum torque and the drive system applies the maximum torque.

In base mode, the machine is operated with $r_{dq} \in \mathrm{MTPA}$ and setting $T_i = T_m$. In field weakening, both operation types are possible. If the set $0 \notin \bar{\Lambda}$, i.e. $\psi_{dq} = [\psi, 0]' \notin \mathcal{I}$ ($|\omega|\psi > \bar{v}_r$), there exists no intersection between MTPA and $\mathrm{br}\,\bar{\Lambda}$. Thus, the machine is operated with $r \in \mathrm{br}\,\bar{\Lambda}$ setting $T_i = 0$.

If $|\omega|\psi \leq \bar{v}_r$, the currents $i_i \in \mathrm{MTPA} \cap \mathrm{br}\,\bar{\Lambda} \neq \emptyset$ (with $i_q > 0$) and define the intersection torque T_i. In other words, the PMSM is operated with $r_{dq} \in \mathrm{MTPA}$ for low reference torques and $r_{dq} \in \mathrm{br}\,\bar{\Lambda}$ for high reference torques, if the intersection between MTPA and $\mathrm{br}\,\bar{\Lambda}$ exists.

In the fourth step, the reference current $r_{dq} \in \mathrm{MTPA}$, i.e. $r_{dq} \in \mathrm{br}\,\bar{\Lambda}$, which produce the reference torque R are computed solving the corresponding quartic equation. Details on the computation of the intersections and finding the reference current r_{dq} are shown in Appendix B.

This procedure computes a current reference vector r_{dq} from a torque reference value R by minimizing the current magnitude. The results are shown in Figure 2.8 in the



(a) Current space

(b) Flux space

Figure 2.10.: PMSM characteristics obtained via measurement (dashed) and using the model with rated parameters (solid, thin)

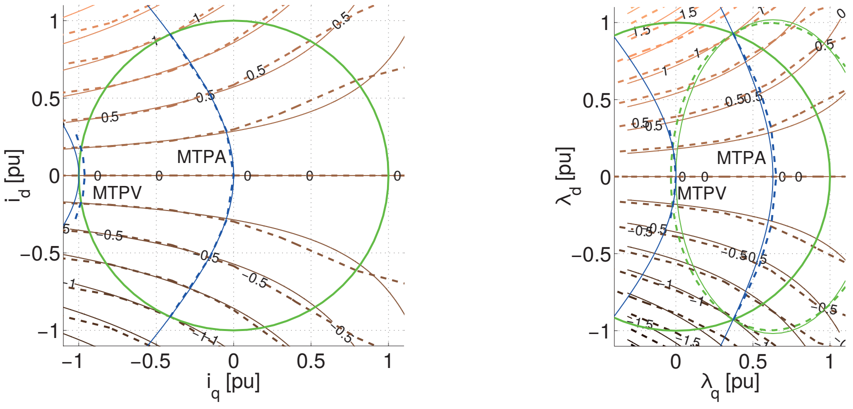
current plane and in Figure 2.9 as function of R. The references r_{dq} denoted with A in Figure 2.8 and Figure 2.9 refer to base mode (constant torque mode), where the operation type is $r \in$ MTPA. Moreover, B and C identify r_{dq} obtained in field weakening.

The machines shown in Figure 2.8(a), i.e. Figure 2.9(a), and Figure 2.8(b), i.e. Figure 2.9(b), have $\omega_m = \infty$. The references r_{dq} denoted with B are obtained in constant power mode and references denoted with C refer to operation in reduced power mode. The machines shown in Figure 2.8(c) and Figure 2.8(d) have $\omega_m < \infty$. Both, B and C refer to r obtained in constant power mode.

2.5. Correction Factors

In this section, the accuracy of the PMSM model is reviewed. As stated previously, the current flux relation $\lambda_{dq} = l(i_{dq})$ is nonlinear in general. Thus, a model based on the linear map $\lambda_{dq} = \mathbf{L}_{dq} i_{dq} + \psi_{dq}$ cannot describe the PMSM exactly. However, it can provide an approximation, which is simple to treat and sufficiently accurate. In practice, a model does not need to describe the machine globally but has to capture the local machine behavior at the desired operation points.

A PMSM comes along with rated parameters, the inductance L_d^r, L_q^r, the PM flux ψ^r, and the pole pairs p^r, which can be used to describe a machine. However, PMSM characteristics (isotorque locus, MTPA and MTPV characteristics) computed



(a) Current space

(b) Flux space

Figure 2.11.: PMSM characteristics obtained via measurement (dashed) and using a model with correction factors computed for the rated operation point (solid, thin)

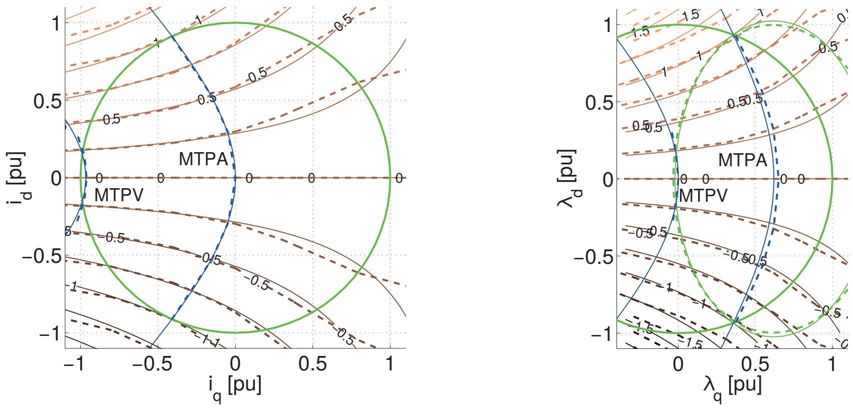
with rated parameters yield often only a rough approximation of the real machine behavior. An example is shown in Figure 2.10, where the computed characteristics are compared to the measured ones.

This model can be improved by targeting the region of desired operation points, which is the left half plane $i_d < 0$ of the current space $i \in \mathcal{I}$, with the improved set of parameters

$$\psi = \rho_\psi \psi^r, \tag{2.11a}$$
$$L_d = \rho_d L_d^r, \tag{2.11b}$$
$$L_q = \rho_q L_q^r, \tag{2.11c}$$
$$p = p^r / \rho_T, \tag{2.11d}$$

where $\rho = [\rho_\psi, \rho_d, \rho_q, \rho_T]' \in \mathbb{R}^4$ are correction factors.

An approach, which often provides good results, is to settle the parameters with respect to the rated operation point with the current $i_r = [i_{dr}, i_{qr}]'$, the flux $\lambda_r = [\lambda_{dr}, \lambda_{qr}]'$ producing the torque T_r. These rated conditions are either previously known or can be settled in laboratory test conditions or during commissioning. The



(a) Current space

(b) Flux space

Figure 2.12.: PMSM characteristics obtained via measurement (dashed) and using a model with correction factors solving a least-squares formulation (solid, thin)

correction factors are obtained solving the linear equation set

$$\begin{bmatrix} 3/2p^r\psi^r i_{qr} & 3/2p^r L_d^r i_{dr} i_{qr} & -3/2p^r L_q^r i_{dr} i_{qr} & -T_r \\ \psi^r i_d & L_d^r(i_{dr}^2 - i_{qr}^2) & -L_q^r(i_{dr}^2 - i_{qr}^2) & \\ \psi^r & L_d^r i_{dr} & & \\ & & L_q^r i_{qr} & \end{bmatrix} \rho = \begin{bmatrix} 0 \\ 0 \\ \lambda_{dr} \\ \lambda_{qr} \end{bmatrix}, \tag{2.12}$$

which is written compactly as $\mathbf{M}\rho = V$ and is solved by $\rho = \mathbf{M}^{-1}V$. The first row settles the torque equation and the second row the MTPA trajectory at the rated operation point. The third and fourth row settles the d and q axis flux-current relation, respectively. The characteristics, computed from the corrected model with rated conditions, are shown in Figure 2.11. This model provides good characterization of the PMSM along the MTPA trajectory, where it is often exact to measurement precision.

For drive systems, where high speed operation is important, other criteria can be added. In particular, an improved correspondence of the MTPV in Figure 2.11(a) is desirable for machines using field weakening. An operation point, which is often known is the demagnetization current i_{d0}, which leads to $L_d i_{d0} = 0$ and defines the short circuit behavior of the machine. Adding this criteria, the equation system becomes

$$\begin{bmatrix} 3/2p^r\psi^r i_{qr} & 3/2p^r L_d^r i_{dr} i_{qr} & -3/2p^r L_q^r i_{dr} i_{qr} & -T_r \\ \psi^r i_d & L_d^r(i_{dr}^2 - i_{qr}^2) & -L_q^r(i_{dr}^2 - i_{qr}^2) & \\ \psi^r & L_d^r i_{dr} & & \\ & & L_q^r i_{qr} & \\ & L_d^r i_{d0} & & \end{bmatrix} \rho = \begin{bmatrix} 0 \\ 0 \\ \lambda_{dr} \\ \lambda_{qr} \\ 0 \end{bmatrix}, \tag{2.13}$$

resulting in an overdetermined equation system with the least squares solution $\rho = \mathbf{M}^{\dagger} V$, where $.^{\dagger}$ denotes the Moore-Penrose pseudoinverse. The characteristics, computed from the corrected model with least squares, are shown in Figure 2.12. Compared to Figure 2.11, an improved MTPV correspondence is obtained.

The approaches provided in this section improve the model locally where the machine is operated. However, the result is still an approximation, which may not characterize well a PMSM with arbitrary cross-saturation. However, modern PMSM design tends to limit the nonlinear behavior [13]. Therefore, it is assumed throughout this text that the PMSM model can be settled such that it describes the real PMSM characteristics with sufficient precision. This section is concluded observing that the PMSM applying the torque T may produce a higher flux magnitude $\|\lambda_{dq}\|$ than that predicted by the model. Thus, it is good practice to introduce the voltage safety factor $\rho_v \in (0, 1]$ as explained in Section 1.5.

Chapter 3.

Inverter

In this chapter, the operating principle of the inverter is shown and the concept of switching function and duty cycle is introduced. Moreover, nonlinear inverter effects are shown and the interface between the gate drivers of the solid-state valves and the control algorithm is defined. A control algorithm stabilizes a desired terminal voltage and the so-called actuation block (direct actuation or modulation) generates a sequence of desired gate signals.

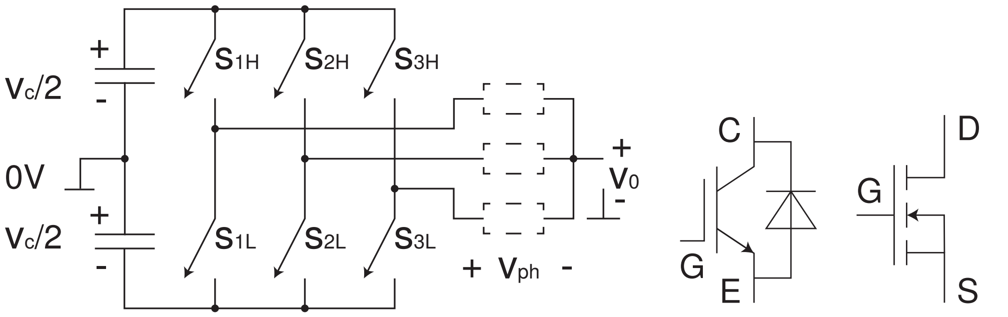
3.1. Model

3.1.1. Switch Model

The two-level VSI topology is shown in Figure 3.1(a) and is realized with solid-state valves [65], which are called the typology of the inverter. Prominent examples of the valves are the insulated gate bipolar transistor (IGBT) with anti-parallel diode, which is shown in Figure 3.1(b), and the metal oxide semiconductor field effect transistor (MOSFET), which is shown in Figure 3.1(c).

For realization of a VSI, the solid-state valves have to be reverse conducting and controllable. Controllable means that the valve conducts or blocks the current flow with respect to a control signal, if a forward voltage is applied. In other words, a positive current between the collector (C) and emitter (E) of the IGBT or the drain (D) and source (S) of the MOSFET, can be allowed or blocked with respect to a control signal applied to the gate (G). If current flow is allowed, the valve is said to be *on*; otherwise, it is said to be *off*. The reverse conducting characteristic is achieved by adding an anti-parallel diode to the IGBT. The MOSFET has an anti-parallel diode inherit in its structure.

Each AC phase is connected to both polarities of the DC link throughout a solid-state valve as it is shown in Figure 3.1(a). The two valves connected to the same phase are



(a) VSI (b) IGBT (c) MOSFET

Figure 3.1.: Voltage source inverter (VSI) topology and example typologies of controllable reverse conducting solid-state valves: the IGBT with anti-parallel diode and the MOSFET

called the *leg* of the inverter. Each of the six valves can either be *on* or *off* leading to 2^6 possible combinations. Some of them are dangerous (short circuit of a leg) or not useful (the resulting AC voltage depends on the current sign). Neglecting those states, the valves of a leg are commanded inverted, i.e. if the upper valve is *on*, the lower one is *off* and vice versa. Thus, each leg has two combinations resulting in 2^3 switching state combinations of the VSI.

Now, the switching state $s_{ph}(t)$ is defined. If the upper valve is *on*, the leg switching state is defined to be 1; otherwise, the leg switching state is 0. Generalizing for the three phase system, the possible switching state of the converter is

$$s_{ph}(t) \in \{0,1\}^3 \subset \mathbb{R}^3. \tag{3.1}$$

The variable is useful to describe the VSI in Figure 3.1(a), where a fictive neutral potential is introduced. From this circuit, the following terminal voltage equation is derived

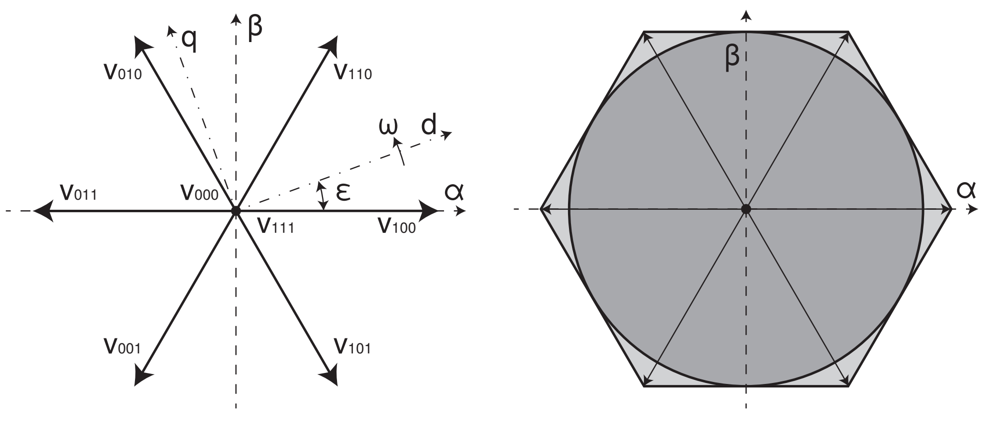
$$v_{ph}(t) = \frac{v_c}{2}(2s_{ph}(t) - \mathbf{1}) - \mathbf{1}v_0(t), \tag{3.2}$$

where v_c is the constant DC link voltage. The zero voltage $v_0(t)$ is computed as

$$v_0(t) = \frac{1}{3}\mathbf{1}'(v_{ph}(t) + \mathbf{1}v_0(t)) = v_c\left(\frac{1}{3}\mathbf{1}'s_{ph}(t) - \frac{1}{2}\right). \tag{3.3}$$

Substituting (3.3) in (3.2) yields

$$v_{ph}(t) = v_c\left(\mathbf{I} - \frac{1}{3}\mathbf{1}\mathbf{1}'\right)s_{ph}(t). \tag{3.4}$$



(a) Set $\mathcal{V}_s$

(b) Set $\mathcal{V}_d$ and $\mathcal{V}$

Figure 3.2.: Terminal voltage $v_{\alpha\beta}$ constraint sets defined by the switching function s_{ph} and the duty cycle d_{ph}

The analysis can be simplified by introducing the $\alpha\beta$ transformation, which yields

$$v_{\alpha\beta}(t) = \mathbf{T}_{\alpha\beta} v_{ph}(t) = v_c \mathbf{T}_{\alpha\beta} \left(\mathbf{I} - \frac{1}{3}\mathbf{11}' \right) s_{ph}(t) = v_c \mathbf{T}_{\alpha\beta} s_{ph}(t) \in \mathcal{V}_s, \tag{3.5}$$

where $\mathcal{V}_s \overset{\text{def}}{=} v_c \mathbf{T}_{\alpha\beta} \circ \{0,1\}^3$ and $\circ$ denotes the convolution. The resulting set is

$$\mathcal{V}_s = v_c \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{v_{000};v_{111}}, \underbrace{\begin{bmatrix} \frac{2}{3} \\ 0 \end{bmatrix}}_{v_{100}}, \underbrace{\begin{bmatrix} \frac{1}{3} \\ \frac{1}{\sqrt{3}} \end{bmatrix}}_{v_{110}}, \underbrace{\begin{bmatrix} -\frac{1}{3} \\ \frac{1}{\sqrt{3}} \end{bmatrix}}_{v_{010}}, \underbrace{\begin{bmatrix} -\frac{2}{3} \\ 0 \end{bmatrix}}_{v_{011}}, \underbrace{\begin{bmatrix} -\frac{1}{3} \\ -\frac{1}{\sqrt{3}} \end{bmatrix}}_{v_{001}}, \underbrace{\begin{bmatrix} \frac{1}{3} \\ -\frac{1}{\sqrt{3}} \end{bmatrix}}_{v_{101}} \right\} \tag{3.6a}$$

$$= v_r \left\{ \underbrace{\begin{bmatrix} 0 \\ 0 \end{bmatrix}}_{v_{000};v_{111}}, \underbrace{\begin{bmatrix} \frac{2}{\sqrt{3}} \\ 0 \end{bmatrix}}_{v_{100}}, \underbrace{\begin{bmatrix} \frac{1}{\sqrt{3}} \\ 1 \end{bmatrix}}_{v_{110}}, \underbrace{\begin{bmatrix} -\frac{1}{\sqrt{3}} \\ 1 \end{bmatrix}}_{v_{010}}, \underbrace{\begin{bmatrix} -\frac{2}{\sqrt{3}} \\ 0 \end{bmatrix}}_{v_{011}} \underbrace{\begin{bmatrix} -\frac{1}{\sqrt{3}} \\ -1 \end{bmatrix}}_{v_{001}}, \underbrace{\begin{bmatrix} \frac{1}{\sqrt{3}} \\ -1 \end{bmatrix}}_{v_{101}}, \right\}. \tag{3.6b}$$

The set $\mathcal{V}_s$ is shown in Figure 3.2(a), where the notation v_{100} refers to the $v_{\alpha\beta}(t)$ voltage vector generated by the switching state $s_{ph}(t) = [1,0,0]'$, etc.

3.1.2. Average Model

The control analysis and design for a system with the integer set constraint $v_{\alpha\beta} \in V_s$ can be difficult and a problem with convex set constraints is often preferable [20].

A power electronic converter can be described using e.g. average modeling [32], where complex but secondary phenomena are neglected. Such a phenomena is the switching behavior of the inverter, which is typically omitted when writing a control model. Thus, the switching state replaced by a relaxed variable called the duty cycle

$$d_{ph}(t) \in [0,1]^3 \subset \mathbb{R}^3, \tag{3.7}$$

which is applied as average of the s_{ph} vector combinations over a sufficiently short time period. Similarly to (3.5), the voltage is

$$v_{\alpha\beta}(t) = v_c \mathbf{T}_{\alpha\beta} d_{ph}(t) \in \mathcal{V}_d, \tag{3.8}$$

where $\mathcal{V}_d \overset{\text{def}}{=} v_c \mathbf{T}_{\alpha\beta} \circ [0,1]^3$ defines the space spanned by $\mathcal{V}_s$. In other words, $\mathcal{V}_d = \operatorname{hull} \mathcal{V}_s$, where hull is the convex hull operator [19], [20]. The resulting set is the polytope defined by

$$\mathcal{V}_d = \left\{ v_{\alpha\beta} \in \mathbb{R}^2 \;\middle|\; \begin{bmatrix} 0 & 1 \\ \sqrt{3}/2 & 1/2 \\ \sqrt{3}/2 & -1/2 \\ 0 & -1 \\ -\sqrt{3}/2 & -1/2 \\ -\sqrt{3}/2 & 1/2 \end{bmatrix} v_{\alpha\beta} \leq \frac{v_c}{\sqrt{3}} \right\} \tag{3.9a}$$

$$= \left\{ v_{\alpha\beta} \in \mathbb{R}^2 \;\middle|\; \begin{bmatrix} 0 & 1 \\ \sqrt{3}/2 & 1/2 \\ \sqrt{3}/2 & -1/2 \\ 0 & -1 \\ -\sqrt{3}/2 & -1/2 \\ -\sqrt{3}/2 & 1/2 \end{bmatrix} v_{\alpha\beta} \leq v_r \right\}. \tag{3.9b}$$

Sometimes, it is advantageous to approximate the set $\mathcal{V}_d$ with the largest two-dimensional ball contained in the set $\mathcal{V} = \operatorname{ball} \mathcal{V}_d$, where ball is the Chebyshev ball operator [19], [20]. This set is defined by

$$\mathcal{V} = \left\{ v_{\alpha\beta} \in \mathbb{R}^2 \mid \|v_{\alpha\beta}\| \leq \frac{v_c}{\sqrt{3}} \right\} \tag{3.10a}$$

$$= \left\{ v_{\alpha\beta} \in \mathbb{R}^2 \mid \|v_{\alpha\beta}\| \leq v_r \right\} \subset \mathcal{V}_d. \tag{3.10b}$$

The stricter constraint set $\mathcal{V}$ is usually preferred over $\mathcal{V}_d$ when operating in the dq reference frame due to its invariance with respect to the Park transformation. Both sets $\mathcal{V}_d$ and $\mathcal{V}$ are shown in Figure 3.2(b).

3.2. Nonlinear Effects

Real solid-state devices introduce a series of non-linear effects in the drive system. These effects tend to distort the desired terminal voltage $v_{\alpha\beta}$. In this section, models are introduced to capture the principal nonlinear effects. However, the models can be further refined and adapted to capture device (IGBT, MOSFET) dependent effects. Such models can be found in [25], [29], [37], [42], [51], [53], [72], [75], [88].

3.2.1. Forward Voltage Drop

A voltage drop occurs in solid state devices when conducting current. The controlled switch, e.g. IGBT, applies $v_S \in \mathbb{R}_+$ and the diode applies $v_D \in \mathbb{R}_+$. Both are typically modeled by a threshold voltage $V_S, V_D \in \mathbb{R}_+$, and a resistance $R_S, R_D \in \mathbb{R}_+$ as shown in Figure 3.3(a). These coefficients are available in the manufacturer data-sheets.

$$v_S(i_S) = V_S + R_S|i_S| \qquad \text{for } i_S > 0, \tag{3.11a}$$

$$v_D(i_D) = V_D + R_D|i_D| \qquad \text{for } i_D > 0. \tag{3.11b}$$

For IGBTs, R_S is typically small and is often neglected or not available. Similarly, V_S is often omitted for MOSFETs. The forward voltage $v_{F,1} \in \mathbb{R}$ applied by an inverter leg setup, e.g. leg 1, is obtained considering Figure 3.5 [37], [88]

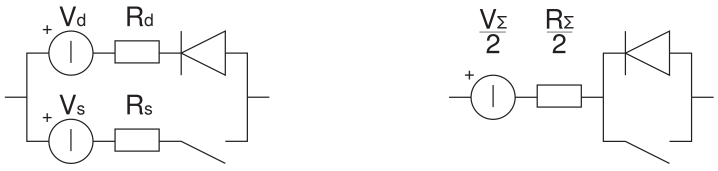
$$v_{F,1} = \begin{cases} -d_1 v_S(i_1) - (1-d_1)v_D(i_1) & \text{for } i_1 > 0 \\ (1-d_1)v_S(i_1) + d_1 v_D(i_1) & \text{for } i_1 < 0 \\ 0 & \text{for } i_1 = 0 \end{cases} \tag{3.12}$$

where the case $i_1 = 0$ has small practical value but is necessary to define $v_{F,1}$ on $\mathbb{R}$. Introducing the sgn operator, the equation can be written as

$$\begin{aligned} v_{F,1} =& \frac{\operatorname{sgn} i_1(\operatorname{sgn} i_1 + 1)}{2}\left(-d_1 v_S(i_1) - (1-d_1)v_D(i_1)\right) \\ &+ \frac{\operatorname{sgn} i_1(\operatorname{sgn} i_1 - 1)}{2}\left((1-d_1)v_S(i_1) + d_1 v_D(i_1)\right). \end{aligned} \tag{3.13}$$

Rearranging, the equation becomes

$$\begin{aligned} v_{F,1} =& -\frac{\operatorname{sgn} i_1}{2}\left(v_S(i_1) + v_D(i_1)\right) \\ &- \frac{\operatorname{sgn}^2 i_1}{2}\left(v_S(i_1) - v_D(i_1)\right)(2d_1 - 1). \end{aligned} \tag{3.14}$$



(a) Switch Model

(b) Approx. Model

Figure 3.3.: Forward voltage drop model

Assuming the case $s_1 = 0$, i.e. $\operatorname{sgn} s_1 = 0$, is never obtained in practice and introducing $V_\Sigma = V_S + V_D$, $V_\Delta = V_S - V_D$, $R_\Sigma = R_S + R_D$, $R_\Delta = R_S - R_D$, the equation becomes

$$v_{F,1} = -\frac{V_\Sigma}{2} \operatorname{sgn} i_1 - \frac{R_\Sigma}{2} i_1 - \left(\frac{V_\Delta}{2} + \frac{R_\Delta}{2} |i_1| \right) (2d_1 - 1) \,. \tag{3.15}$$

The voltage drop models often satisfy, $v_S(i_1) + v_D(i_1) \gg v_S(i_1) - v_D(i_1)$. Moreover, the model needs to be accurate primarily at voltages with small magnitude, i.e. duty cycles $d_1 \approx 0.5$. For larger magnitudes, the voltage drop has a limited impact. Thus, the model is often approximated [75], [88] with the model shown in Figure 3.3(b)

$$v_{F,1} = -\frac{V_\Sigma}{2} \operatorname{sgn} i_1 - \frac{R_\Sigma}{2} i_1. \tag{3.16a}$$

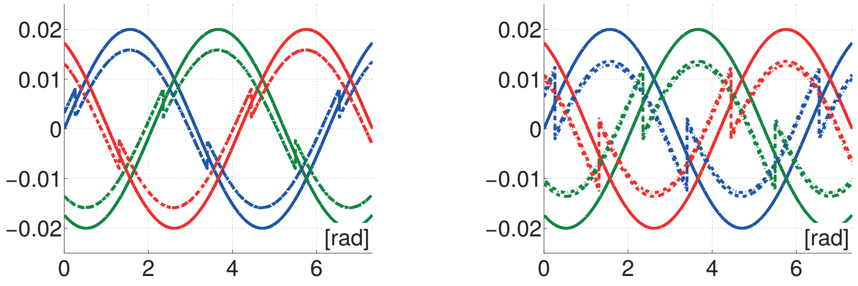
For the three phase system, the forward voltage drop vector $v_{F,ph} = [v_{F,1}, v_{F,2}, v_{F,3}]' \in \mathbb{R}^3$ is defined as

$$v_{F,ph} = -\frac{V_\Sigma}{2} \operatorname{sgn} i_{ph} - \frac{R_\Sigma}{2} i_{ph}, \tag{3.17}$$

which can be represented in the $\alpha\beta$ reference frame as $v_{F,\alpha\beta} = \mathbf{T}_{\alpha\beta} v_{F,ph}$ and yield

$$v_{F,\alpha\beta} = -\frac{V_\Sigma}{2} i_{\alpha\beta,sgn} - \frac{R_\Sigma}{2} i_{\alpha\beta}, \tag{3.18}$$

where $i_{\alpha\beta,sgn} = \mathbf{T}_{\alpha\beta} \operatorname{sgn} i_{ph}$. Neglecting the cases $i_1 = 0$, i.e. $\operatorname{sgn} i_1 = 0$, the vector $i_{\alpha\beta,sgn}$ belongs to a set, which is similar to $\mathcal{S}_{\alpha\beta}$ in Figure 3.2(a) and each active vector (with nonzero magnitude) has length 4/3. Typical distortions due to the voltage drop are shown in Figure 3.4(a) for voltages with low magnitude imposing a sinusoidal current. The exact (dashed trajectory) and approximated (dash-dotted trajectory) distortion model produce an overlaying trajectory.



(a) Voltage drop

(b) Interlock time

Figure 3.4.: Voltage distortions imposing a sinusoidal current with $\pi/12$ phase lag: ideal voltage (continuous) with magnitude $\|v_{\alpha\beta}\|_2/v_c = 0.01$ and real voltage with exact (dashed) and approximated (dash-dotted) model

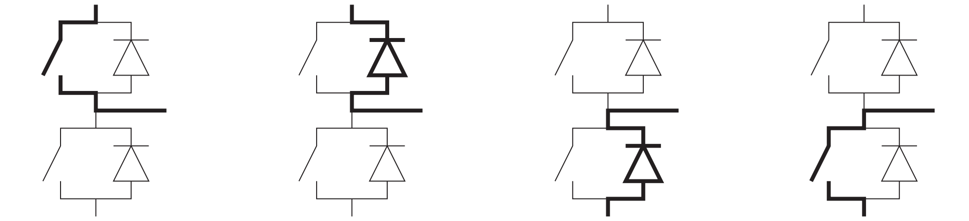
3.2.2. Interlock Time

The real switching behavior of solid-state devices introduces voltage distortions. Switching transients are shown in Figure 3.6 referring to leg 1. The voltages v_{1H}, v_{1L} and currents i_{1H}, i_{1L} are the values on, i.e. in, the devices S_{1H} and S_{1L} in Figure 3.1(a). The transient from the switching state $s_1 = 1$ to $s_1 = 0$ for $i_1 > 0$ and $i_1 < 0$ are shown in Figure 3.6(a) and Figure 3.6(b), respectively. In other words, the changes from the case in Figure 3.5(a) to Figure 3.5(c) and Figure 3.5(b) to Figure 3.5(d) are shown. The transients from $s_1 = 0$ to $s_1 = 1$ are similar.

A real controlled switch introduces a delay $t_{d,on}$, $t_{d,off}$ (for *on* and *off* switching) when actuating a gate command. Moreover, it does not change its state instantaneously but requires some time $t_{r,on}$, $t_{r,off}$ until the voltage on the switch raises to $0V$, i.e. the DC-link voltage v_c. Consequently, S_{1H} and S_{1L} cannot change the switching state at the same instant due to the risk of a DC link short-circuit. Thus, an interlock time T_i (sometimes also called dead-time) is introduced. During this interval an

Table 3.1.: Switching delays referring to leg 1

Transient	Current	Delay
$s_1 = 1$ to $s_1 = 0$	$i_1 > 0$	$T_{off} \overset{\text{def}}{=} t_{d,off} + t_{s,off}/2$
$s_1 = 1$ to $s_1 = 0$	$i_1 < 0$	$T_{on} \overset{\text{def}}{=} T_i + t_{d,on} + t_{s,on}/2$
$s_1 = 0$ to $s_1 = 1$	$i_1 > 0$	T_{on}
$s_1 = 0$ to $s_1 = 1$	$i_1 < 0$	T_{off}



(a) $s_1 = 1$, $i_1 > 0$ (b) $s_1 = 1$, $i_1 < 0$ (c) $s_1 = 0$, $i_1 > 0$ (d) $s_1 = 0$, $i_1 < 0$

Figure 3.5.: Active components of leg 1

off command $s_{1H} = 0$, $s_{1L} = 0$ is applied to both switches to ensure that the switch, which was previously *on*, has fully turned *off* before turning *on* the other one.

The turn *on* and *off* delays are summarized in Table 3.1. These delays add an undesired duty cycle $d_{S,1}$ to the desired d_1, which can be written as

$$d_{S,1} = \begin{cases} n_{10,1}T_{\mathit{off}}/T_s - n_{01,1}T_{on}/T_s & \text{for } i_1 > 0 \\ n_{10,1}T_{on}/T_s - n_{01,1}T_{\mathit{off}}/T_s & \text{for } i_1 < 0 \\ 0 & \text{for } i_1 = 0 \end{cases} \tag{3.19}$$

where $n_{10,1} \in \mathbb{N}$ and $n_{01,1} \in \mathbb{N}$ are the number of $s_1 = 1$ to $s_1 = 0$ and $s_1 = 0$ to $s_1 = 1$ transients during the actuation interval T_s, respectively. Introducing the sgn operator, the equation can be written as

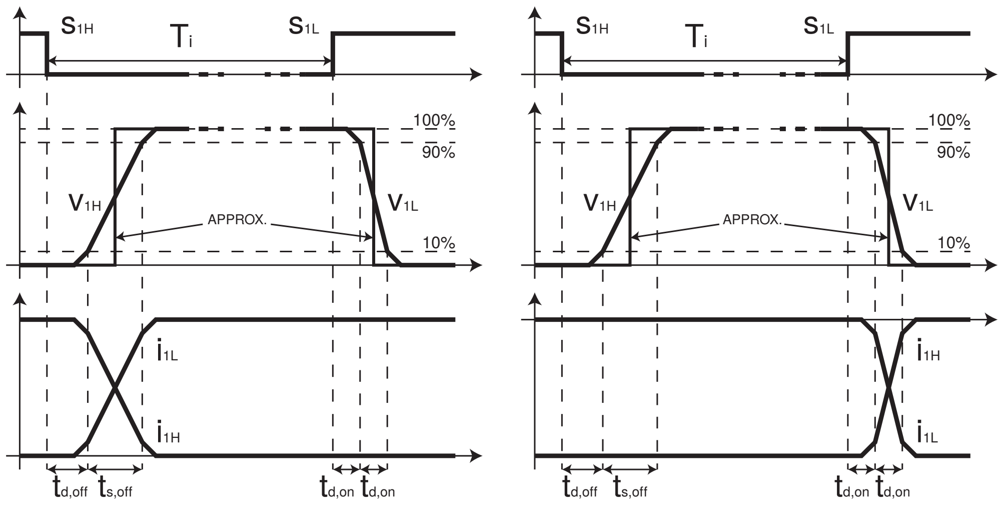
$$\begin{aligned} d_{S,1} = & \frac{\operatorname{sgn} i_1(\operatorname{sgn} i_1 + 1)}{2}\left(n_{10,1}\frac{T_{\mathit{off}}}{T_s} - n_{01,1}\frac{T_{on}}{T_s}\right) \\ & + \frac{\operatorname{sgn} i_1(\operatorname{sgn} i_1 - 1)}{2}\left(n_{10,1}\frac{T_{on}}{T_s} - n_{01,1}\frac{T_{\mathit{off}}}{T_s}\right). \end{aligned} \tag{3.20}$$

Rearranging, the equation becomes

$$\begin{aligned} d_{S,1} = & -\frac{\operatorname{sgn} i_1}{2}\left(n_{10,1} + n_{01,1}\right)\frac{T_{on} - T_{\mathit{off}}}{T_s} \\ & + \frac{\operatorname{sgn}^2 i_1}{2}\left(n_{10,1} - n_{01,1}\right)\frac{T_{on} + T_{\mathit{off}}}{T_s} \end{aligned} \tag{3.21}$$

Assuming the case $s_1 = 0$, i.e. $\operatorname{sgn} s_1 = 0$, is purely mathematical and introducing $n_{\Sigma,1} = n_{10,1} + n_{01,1}$, $n_{\Delta,1} = n_{10,1} - n_{01,1}$, $T_\Sigma = T_{on} + T_{\mathit{off}}$, and $T_\Delta = T_{on} - T_{\mathit{off}}$, the equation becomes

$$d_{S,1} = -\frac{n_{\Sigma,1}}{2}\frac{T_\Delta}{T_s}\operatorname{sgn} i_1 + \frac{n_{\Delta,1}}{2}\frac{T_\Sigma}{T_s}. \tag{3.22}$$



(a) $i_1 > 0$

(b) $i_1 < 0$

Figure 3.6.: Switching transients of leg 1 from $s_1 = 1$ to $s_1 = 0$

This equation can be simplified, if a periodic switching sequence is employed. Modulation schemes, e.g. PWM, typically use $n_{10,1} = n_{01,1}$ (within T_s), which yields [88]

$$d_{S,1} = -f_{sw}T_\Delta \operatorname{sgn} i_1, \tag{3.23}$$

where f_{sw} is the switching frequency. Moreover, a pure interlock model $T_\Delta \approx T_i$ can be considered, if $T_i \gg t_{d,on} - t_{d,off} + (t_{s,on} - t_{s,off})/2$.

For the three phase system, the switching transient duty cycle $d_{S,ph} = [d_{S,1}, d_{S,2}, d_{S,3}]' \in \mathbb{R}^3$ is defined as

$$d_{F,ph} = -f_{sw}T_\Delta \operatorname{sgn} i_{ph}, \tag{3.24}$$

which can be written as $\alpha\beta$ voltage vector [88]

$$v_{S,\alpha\beta} = v_c\mathbf{T}_{\alpha\beta}d_{S,ph} = -v_c f_{sw}T_\Delta i_{\alpha\beta,sgn}. \tag{3.25}$$

Typical distortions due to the switching transients are shown in Figure 3.4(b) for voltages with low magnitude and a sinusoidal current. The exact model is shown as a dashed trajectory. The approximation $T_\Delta \approx T_i$ is shown as dash-dotted trajectory.

3.3. Actuation Scheme

In this section, different ways to actuate the terminal voltage $v_{\alpha\beta}$ to the plant by the inverter are elaborated. The terminal voltage $v_{\alpha\beta}$ is typically the plant input of a control system. Thus, the control system decides a $v_{\alpha\beta}$ to actuate. However, the inverter requires either a switching state s_{ph} to actuate directly a voltage vector or a duty cycle d_{ph}, when a modulator is used.

Mathematically, the voltage $v_{\alpha\beta}$ is the result of (3.5), i.e. (3.8). Since these equations involve the Clarke transformation, they cannot be inverted (s_{ph}, i.e. d_{ph}, has dimension three but $v_{\alpha\beta}$ has dimension two). In practice, strategies are used to compute s_{ph}, i.e. d_{ph} such that (3.5), i.e. (3.8) hold. The additional degree of freedom is used to the advantage of the system e.g. to decrease the switching frequency or to improve the current quality.

Direct actuation considers applying voltage vectors $v_{\alpha\beta} \in \mathcal{V}_s$ to the PMSM. The result is a switching state $s_{ph} \in \{0, 1\}^3$, which can be directly applied to the inverter. In contrast, a voltage $v_{\alpha\beta} \in \mathcal{V}_d$ is applied throughout a *modulation scheme* to the inverter. Typical examples are the *Pulse Width Modulation* (PWM) and *Space Vector Modulation* (SVM). According to the PWM theorem [44], there exists a $d_{ph} \in [0, 1]^3$ for each vector $v_{\alpha\beta} \in \mathcal{V}_d$. Once d_{ph} is known, gate signals are generated using PWM, i.e. intersecting d_{ph} with a (triangular or sawtooth) carrier signal [65].

Several *modulation schemes* have been proposed in literature [4], [41], [50], [76]. Each of them determines d_{ph} with advantages and drawbacks with respect to the current quality (THD) and the switching frequency f_{sw}. In this text, two modulation techniques, which are known to provide good results, are introduced [102]. The Symmetric Space Vector Modulation (SSVM) divides symmetrically the passive states and is known for obtaining good current quality. Discontinuous Space Vector Modulation (DSVM) avoids switching of one inverter leg and is known for obtaining a low switching frequency and switching losses.

3.3.1. Direct Actuation: Actuation of a Voltage Vector

Direct actuation is a relatively simple actuation scheme. Fixing $v_{\alpha\beta} \in \mathcal{V}_s$, the switching function $s_{ph} \in \{0, 1\}^3$ is uniquely defined if $v_{\alpha\beta}$ is an active vector $v_{\alpha\beta} \neq 0$. However, the passive vector $v_{\alpha\beta} = 0$ can be applied using $s_{ph} = [0, 0, 0]'$ or $s_{ph} = [1, 1, 1]'$. Thus, direct actuation requires a strategy for selecting the passive voltage.

Since the previously actuated vector $s_{ph}[k-1]$ is known, a typical strategy is to minimize the number of overall switching transitions, which cause the switching power loss. The number of switching transitions are defined as follows

$$n_{sw}[k] \overset{\text{def}}{=} \left\| s_{ph}[k] - s_{ph}[k-1] \right\|_1 , \tag{3.26}$$

and the number of switching transitions for more periods is $\sum_{j=0}^{N-1} n_{sw}[k+j]$, where $N \in [1, \infty)$ is the horizon under consideration.

Proposition 3.1. *Choose for all passive vectors $v_{\alpha\beta}[k]$ with $k \in \mathbb{N}_+$ the switch state $s_{ph}[k] = [0,0,0]'$ or $s_{ph}[k] = [1,1,1]'$ such that $n_{sw}[k]$ is minimum, then the overall number of switching transitions*

$$\sum_{j=1}^{\infty} n_{sw}[j] \tag{3.27}$$

is minimum.

Proof. The proof is by induction. Let $s_{ph}[0]$ be any vector, then the case $M = 1$ is true since $\sum_{j=1}^{M} n_{sw}[j] = n_{sw}[1]$ is minimum choosing $s_{ph}[1]$ according the ruling of the proposition. Now, the proposition is proven if it holds also for the case $M + 1$. Let M define the length of any sequence of active (a) and passive (p) switch states $s_{ph}[0], \ldots, s_{ph}[M]$, which has a minimum number of switch transitions $\sum_{j=1}^{M} n_{sw}[j]$. If any vector $s_{ph}[M+1]$ is added to this sequence by ruling of the proposition, the number of switch transitions $\sum_{j=1}^{M+1} n_{sw}[j]$ must still be minimum.

Since any active switch state is defined by the terminal voltage vector, its switching cannot be optimized. Thus, not the entire sequence needs to be considered but only the subsequence back to the last active switch state. Considering e.g. the sequence p, p, a, p, the switching of p, p, a is minimum by assumption and only the subsequence a, p needs to be considered when adding a switch state. Consequently, the proposition is true if the last vector of the existing sequence is active, e.g. p, a, a, since adding any vector results in a case similar to $N = 1$.

Moreover, the inverter does not switch if multiple passive vectors are applied in a row, e.g. the sequence a, p, p, a and a, p, a produce the same number of switch transitions, and the problem of showing, which passive vector yields the minimum number of overall switch transitions, remains the same. Thus, the problem reduces to show that the overall switching transitions of the sequence $a, p, a/p$ (the last vector is the active or passive vector to add) are minimum if the passive vectors are chosen according to the proposition. Thus, the following six cases are obtained:

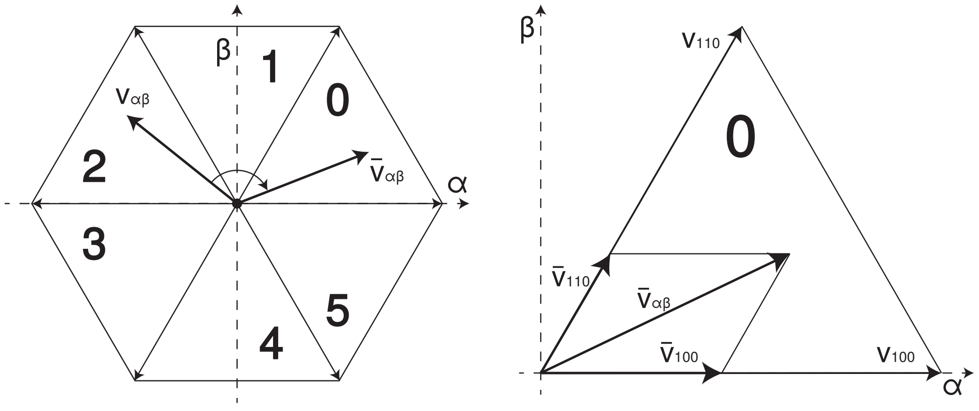
$$[1,0,0]' - [0,0,0]' - [0,0,0]' \tag{3.28a}$$
$$[1,0,0]' - [0,0,0]' - [0,0,1]' \tag{3.28b}$$
$$[1,0,0]' - [0,0,0]' - [0,1,1]' \tag{3.28c}$$
$$[1,1,0]' - [1,1,1]' - [0,0,0]' \tag{3.28d}$$
$$[1,1,0]' - [1,1,1]' - [0,1,1]' \tag{3.28e}$$
$$[1,1,0]' - [1,1,1]' - [1,1,1]' \tag{3.28f}$$

In (3.28a) and (3.28f), the vector to add (last vector) is a passive vector, which is chosen to be the same as the last vector of the existing sequence (middle vector). Thus, the number of switching transitions of the sequence is minimum. In (3.28b) and (3.28e), the active vector to add (last vector) has the same number of *on* states than the last active vector (first vector). Thus, choosing the passive vector according to this proposition leads to two switching transitions (choosing the opposite passive vector leads to four). In (3.28c), i.e. (3.28d), the active vector to add (last vector) has one more, i.e. one less, of *on* states than the last active vector (first vector). Thus, both passive vectors lead to the same number (three) of switching transitions. □

The Proposition 3.1 states that the overall number of switching transitions is minimized if the passive switching state is chosen, which can be reached from the previous state with the fewest switching transitions. Consequently, there exists a simple rule how to choose $s_{ph}[k], s_{ph}[k+1], \dots$ once a sequence of plant inputs, i.e. voltage vectors $v_{\alpha\beta}[k], v_{\alpha\beta}[k+1], \dots$ is defined. The main advantage of this result is that the actuation scheme is proper. The global switching behavior can be optimized making decisions based on actual (and past) information. Information on the future evolution of the switching sequence is not necessary to choose an optimal (passive) switch state. However, a control algorithm can further optimize the number of switching transitions. Such an algorithm can optimize the location of the states and the necessary plant inputs, i.e. the active and passive voltage vectors.

3.3.2. Symmetric Space Vector Modulation (SSVM)

The input of a modulator is the voltage vector $v_{\alpha\beta}$, which is defined by its magnitude $\|v_{\alpha\beta}\|$ and its angle $\angle v_{\alpha\beta}$ in polar coordinates. The vector is required to satisfy $v_{\alpha\beta} \in \mathcal{V}_d$. If this requirement fails, e.g. due to numerical precision, the magnitude $\|v_{\alpha\beta}\|$ is reduced such that $v_{\alpha\beta} \in \mathcal{V}_d$.



(a) Sectors (b) Vector cycle

Figure 3.7.: SVM sectors and computation of vector duty cycles

The vector $v_{\alpha\beta}$ lies in one of the six sectors ($sec \in \{0, ..., 5\}$) defined by Figure 3.7(a). To simplify the treatment, $v_{\alpha\beta}$ is rotated into the sector zero and defines $\bar{v}_{\alpha\beta}$, with $\|\bar{v}_{\alpha\beta}\| = \|v_{\alpha\beta}\|$ and $\angle\bar{v}_{\alpha\beta} = \angle v_{\alpha\beta} - sec\ \pi/3$.

The vector $\bar{v}_{\alpha\beta}$ is obtained as combination of v_{100}, v_{110}, and a passive vector e.g. v_{000}. The components are obtained using the law of sine as shown in Figure 3.7(b)

$$\|\bar{v}_{100}\| = \frac{\|\bar{v}_{\alpha\beta}\|}{\sin(2\pi/3)} \sin(\pi/3 - \angle\bar{v}_{\alpha\beta}) = \frac{2}{\sqrt{3}} \|\bar{v}_{\alpha\beta}\| \sin(\pi/3 - \angle\bar{v}_{\alpha\beta}), \tag{3.29a}$$

$$\|\bar{v}_{110}\| = \frac{\|\bar{v}_{\alpha\beta}\|}{\sin(2\pi/3)} \sin(\angle\bar{v}_{\alpha\beta}) = \frac{2}{\sqrt{3}} \|\bar{v}_{\alpha\beta}\| \sin(\angle\bar{v}_{\alpha\beta}). \tag{3.29b}$$

The *on*-time t_{100} of the vector v_{100} in the averaging period T_s (typically the sampling period) is proportional to the length of $\bar{v}_{100}$ and yields the vector duty cycles

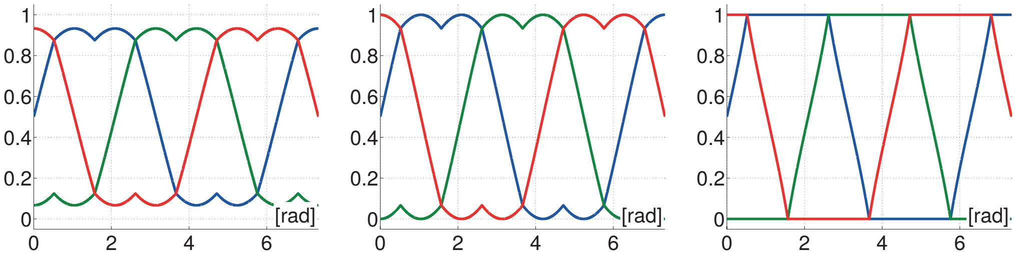
$$d_{100} = \frac{t_{100}}{T_s} = \frac{\|\bar{v}_{100}\|_2}{\|v_{100}\|_2} = \|\bar{v}_{100}\| \frac{3}{2v_c}, \tag{3.30a}$$

$$d_{110} = \frac{t_{110}}{T_s} = \frac{\|\bar{v}_{110}\|_2}{\|v_{110}\|_2} = \|\bar{v}_{110}\| \frac{3}{2v_c}. \tag{3.30b}$$

The remaining time of T_s, a passive vector is applied with

$$d_{000} = 1 - d_{100} - d_{110}. \tag{3.31}$$

Now, the duty cycle d_{ph} is computed and actuated to the inverter by PWM. Since d_{ph} is not uniquely defined, several schemes exist to determine it. SSVM divides the



(a) $\|v_{\alpha\beta}\|/v_c = 1/2$ (b) $\|v_{\alpha\beta}\|/v_c = 1/\sqrt{3}$ (c) $\|v_{\alpha\beta}\|/v_c = max$

Figure 3.8.: SSVM duty cycle d_{ph} for different modulation ratios $\|v_{\alpha\beta}\|/v_c$

passive cycle d_{000} and applies half of it at the beginning and half at the end of the sampling period. The resulting duty cycles, which define SSVM, are summarized in Table 3.2.

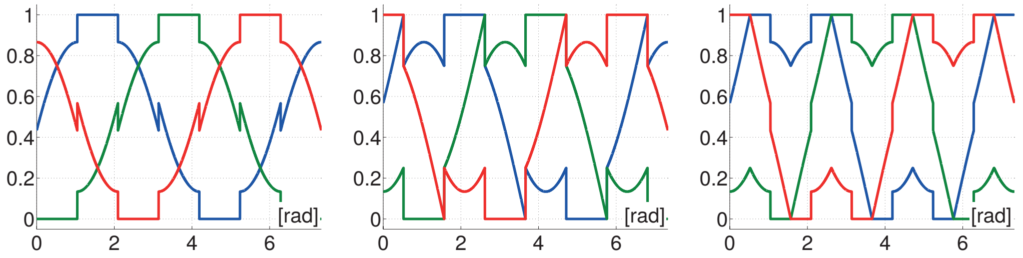
The duty cycle d_{ph} generated by SSVM is shown in Figure 3.8 for different ratios $\|v_{\alpha\beta}\|/v_c$. For $\|v_{\alpha\beta}\|/v_c \leq 1/2$ (Figure 3.8(a)) the modulation is said to be in the linear range [65]. For $1/2 < \|v_{\alpha\beta}\|/v_c \leq 1/\sqrt{3}$ (Figure 3.8(b)), the injection of a third harmonic (and its multiples) is required [65].

For $\|v_{\alpha\beta}\|/v_c > 1/\sqrt{3}$ the system is said to work in over-modulation. A larger harmonic content is injected and the voltage and current quality (THD) decreases [65]. A vector $v_{\alpha\beta}$ rotating with constant velocity ω and maximum magnitude $\|v_{\alpha\beta}\|/v_c = max$, generates trapezoidal d_{ph} as shown in Figure 3.8(c), with the fundamental magnitude $6/\pi^2$ [93].

It is well known [44], [65], [93], that the fundamental magnitude can be further increased to $2/\pi$ where rectangular d_{ph} are generated. However, $v_{\alpha\beta}$ rotates with non-constant ω, which further decreases the voltage and current quality. In particular, $v_{\alpha\beta}$

Table 3.2.: Duty cycle computation

sec	d_{ph}
0	$[0.5d_{000} + d_{100} + d_{110},\ 0.5d_{000} + d_{110},\ 0.5d_{000}]'$
1	$[0.5d_{000} + d_{100},\ 0.5d_{000} + d_{100} + d_{110},\ 0.5d_{000}]'$
2	$[0.5d_{000},\ 0.5d_{000} + d_{100} + d_{110},\ 0.5d_{000} + d_{110}]'$
3	$[0.5d_{000},\ 0.5d_{000} + d_{100},\ 0.5d_{000} + d_{100} + d_{110}]'$
4	$[0.5d_{000} + d_{110},\ 0.5d_{000},\ 0.5d_{000} + d_{100} + d_{110}]'$
5	$[0.5d_{000} + d_{100} + d_{110},\ 0.5d_{000},\ 0.5d_{000} + d_{100}]'$



(a) $\varphi = 0$ (b) $\varphi = \pi/4$ (c) $\varphi = \pi/2$

Figure 3.9.: DSVM duty cycle d_{ph} for different phase lags φ and $\|v_{\alpha\beta}\|_2/v_c = 1/2$

"jumps" from vertex to vertex of $\mathcal{V}_d$ when rectangular d_{ph} with maximum fundamental magnitude are generated. Thus, this operation is not appreciated in general and it is not emphasized in this text.

3.3.3. Discontinuous Space Vector Modulation (DSVM)

DSVM uses the property that any zero component $d_0 \in \mathbb{R}$ can be added to d_{ph} as long $d_{ph} + d_0 \in \mathcal{V}_d$. Due to shortness, DSVM is explained based on SSVM but it can also be written independently. The switching losses of an inverter leg are approximately proportional to the DC-link voltage v_c and increase with the absolute value of the current i_{123} in the leg [84]. However, switching losses are independent of the terminal voltage. In order to reduce the losses, it is preferable to not switch the leg carrying the highest (absolute) current.

After identifying the leg with the highest current magnitude, d_{ph} is modified such that the duty cycle in that leg is 0 or 1 and therefore it is not switched. However, this strategy requires that the duty cycle of the leg with the highest current is either minimum or maximum of the three. This depends on the phase lag φ of the current and is not always true. Is this is not the case, switching can be avoided in the leg with the second highest current magnitude, where as a result the duty cycle is either minimum or maximum. As a result the switching frequency decreases by approximately 33% and the switching losses decrease by a similar amount since switching is avoided in phases with high currents.

This procedure assumes that the phase currents are clearly detectable despite of measurement noise, etc. This is generally difficult for currents with low magnitude, e.g. $\|i_{123}\|_2 < 1\% I_r$. Thus, DSVM is avoided in this conditions and the modulation falls back to SSVM increasing the switching frequency. However, this approach is acceptable since the switching (and conduction) losses are particularly low in these conditions.

The duty cycle d_{ph} generated by DSVM is shown in Figure 3.9 for the ratios $\|v_{\alpha\beta}\|_2/v_c = 1/2$. The modulation depends on the phase lag of the current, ϕ. The duty cycle d_{ph} generated for $\phi = 0$ is shown in Figure 3.9(a), $\phi = \pi/4$ is shown in Figure 3.9(b), and $\phi = \pi/2$ is shown in Figure 3.9(a). At high $\|v_{\alpha\beta}\|_2/v_c$ ratios, DSVM behaves similar to SSVM.

Part II.

Model Predictive Control (MPC)

Introduction

Drive systems use a control architecture to apply a desired torque to a rotating shaft, e.g. [9], [27], [34], [47], [49], [87], [89]. In Part I, the control framework is introduced to reduce the complexity of dynamic control via static mappings. The control problem simplifies from actuating a torque reference vector by selecting gate signals into actuating a state, i.e. current, reference vector by choosing a terminal voltage. Closed-loop control is provided using plant feedback with a typical drive system sensor set. It contains the three phase current and DC link voltage measurement, as well as the (electric) rotor speed and angle, which are available either through measurement via encoder or as estimate via observer. Throughout this text, the dynamic control layout, which is shown in Figure II.1, is used. The Clarke, Park, and current-flux transformation are used to map references and measurements into the desired spaces. They are conceptually part of the control framework and are marked gray. The observer removes, i.e. reduces, the effect of delays, signal noise, and model uncertainties [80]–[82]. It is based on the *dq* reference frame, where integration can be used to remove steady-state errors. It can be designed and computed independently from the controller due to the separation principle [40], [101]. Details on the observer design and implementation are treated in Appendix C.

The Part II of this text considers the control design and implementation. The stator flux state-space model in the stationary $\alpha\beta$ reference frame is used to describe the system due to its simplicity. However, a stator flux measurement is not available in most drive systems. Thus, the model uses a *virtual flux*, which is computed from the current measurement via transformation. Since the same transformation is applied to the (current) reference, they cancel out and the controller behaves like a current controller. In other words, the controller tends to remove the current error but the real flux error can be arbitrarily large, if the current-flux map is inexact. For simplicity, the linearized current-flux map (1.10) is used throughout this text. This relation can be replaced with more accurate nonlinear formulations, e.g. (1.11) or (1.12) to capture saturation or cross-saturation (a replacement requires some mathematical properties of the nonlinear maps). The tracking problem is transformed into a regulation problem with a change of states from the stator flux to the stator flux error. The resulting control problem is solved by a linear combination of a *feedforward* and *feedback* controller. Both are designed considering system

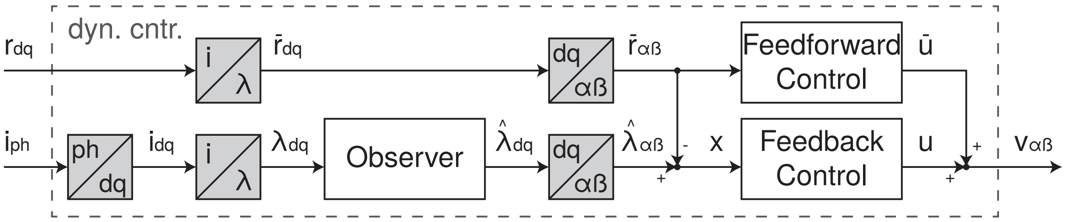


Figure II.1.: Dynamic control scheme with static mappings (gray) and dynamic blocks (white).

constraints. State (current) constraints are considered when generating the reference values but are neglected in dynamic operation since transient violation of these constraints is typically acceptable. Thus, an *input-constraint* dynamic system is obtained. Choosing the $\alpha\beta$ reference frame is advantageous since the input constraint sets (3.6), i.e. (3.9), are constant (they are rotating in the *dq* reference frame) but shifted due to the feedforward controller.

The feedback control is implemented using *Model Predictive Control* (MPC) [19], [61]. MPC solves a *Constrained Finite Time Optimal Control* (CFTOC) problem at each sampling time instant. There, the control goal is defined as *cost function* and the behavior of the constrained dynamic system is optimized over N future time steps, which is called the *prediction horizon*. If a solution to the CFTOC problem exists, it is solved by the *optimal* state and input sequence, which lead to the minimum cost among all *feasible* sequences. MPC implements the first element of the optimal input sequence as feedback control action. At the next sampling instant, the CFTOC is solved with updated measurements over the shifted horizon (thus, MPC is also named *Receding Horizon Control* (RHC)). MPC is capable to work with both input constraint sets. Whether CCS or FCS, i.e. PWM or direct actuation, is used results in significantly different control properties, which are beneficial for different applications. Differences include a constant or variable switching frequency, sampling to switching frequency ratio, current quality, and acoustic noise [79].

The main MPC stability theorem [19], [64] is conservative and typically difficult to apply to drive systems due to computation constraints. Thus, the Lypunov-based MPC approach [26], [68] is introduced to drive systems in order to ensure stability, where a *Control Lyapunov Function* (CLF) stability constraint is used. This constraint ensures that at least the first of the optimal input sequence decreases the value of a Lyapunov function. If the resulting CFTOC is *persistently* feasible (feasible for all time), the resulting closed-loop system is stable. The MPC stability properties depend on the input constraint set. Using CCS-MPC, the switching behavior of the drive system is transparent to the control system. As a result the (average) control error can be brought to the origin and the system is shown to be stable in the sense of Lyapunov. In contrast, FCS-MPC takes the switching behavior into account. It is able to reduce the value of a Lyapunov function but within a (well-defined) sublevel set, the FCS does not contain any input to further decrease the Lyapunov function. Thus, FCS-MPC is able to steer the control error towards a neighborhood of the origin where the error remains ultimately bounded and is shown to be set-stable.

Power electronic and drive systems require fast implementations of the control code. Code is typically executed on embedded hardware with limited computation capabilities and needs to provide a result, i.e. an updated input, within a (small) sampling period. Consequently, the CFTOC must be designed to be sufficiently simple, which is typically achieved using a prediction horizon of a few steps. Moreover, solver algorithms need to be computationally efficient. The CCS-MPC CFTOC is a (convex) linear or quadratic programming problem (neglecting CLF, which lead to quadratic constraints). For this class of problems, efficient solving strategies exist and are well documented in literature. An interesting approach for fast sampled input constrained systems is e.g. the fast gradient method [83]. In this text, the proposed CCS-MPC (with horizon $N = 1$) is shown to be solvable by an analytic algorithm, which yields the exact solution to the CFTOC efficiently. The FCS-MPC CFTOC is a mixed-integer linear or quadratic programming problem and consequently more demanding to solve. It is typically solved by enumeration [80]–[82]. In this text, the enumeration procedure is combined with *branch-and-bound*, i.e. *branch-and-cut*, techniques to improve the computational efficiency.

The Part II of this text is divided into three chapters. In Chapter 4, a simple nonlinear controller is designed and implemented for the constrained dynamic system using PWM actuation. The resulting closed-loop system is shown to be stable using Lyapunov functions, which simplifies the problem of finding CLF for CCS-MPC. The nonlinear controller is used to validate the control framework and the dynamic control architecture. It acts as a benchmark reference for the closed-loop control behavior in dynamic operation (speed and torque reference step) and in steady-state conditions (location of operation points and current quality). The evaluation is carried out on a Software-in-the-Loop (SiL) platform and on an experimental test bench, which are described in detail in Appendix D. The CCS-MPC and FCS-MPC is described in Chapter 5 and Chapter 6, respectively. First, the dynamic system and the constraints are described and analyzed for their stabilizability using CLF. Then, the proposed CFTOC and efficient algorithms to solve the optimization problems are described. Both CCS-MPC and the FCS-MPC are evaluated in the same conditions as the nonlinear controller.

Chapter 4.

Nonlinear Control

In this chapter, a nonlinear controller is introduced for drive systems with convex control set, i.e. using a modulator. This controller is shown to yield a stable constrained closed loop system, which is a sufficient condition that the underlying constrained system is stabilizable. In other words, there always exists an input such that the state, i.e. control error, converges towards the origin and this input satisfies the system constraints. Efficient algorithms are presented such that the nonlinear controller can be implemented on embedded control hardware with typical real time requirements of power electronic systems. The nonlinear controller is evaluated in simulation and on an experimental test bench and provides the benchmark reference for further (model predictive) control developments.

4.1. System Dynamics

For the control design, the drive system model in the $\alpha\beta$ flux space is used, which provides a simple model. The dynamics is described based on the discrete-time state-space formulation (1.19)

$$\lambda_{\alpha\beta}[k+1] = \lambda_{\alpha\beta}[k] + T_s v_{\alpha\beta}[k], \tag{4.1}$$

which is an integrator with two dimensions. Compared to (1.19), secondary effects are not considered and the following is assumed.

Assumption 4.1. Disturbances (resistive voltage drops, inverter nonlinear effects, etc.) are compensated externally, e.g. by an observer.

The control goal is to track a reference vector. The original reference is a current reference r_{dq}, which defines a flux reference $\bar{r}_{dq}$ using the relation (1.10)

$$\bar{r}_{dq} = \mathbf{L} r_{dq} + \psi_{dq}. \tag{4.2}$$

The reference is set externally, e.g. by the reference generation procedure outlined in in Section 2.4. For simplicity, it is not allowed to be arbitrarily fast varying making the following assumption.

Assumption 4.2. The current reference r_{dq} is slow varying with respect to the sampling time T_s such that $r_{dq}[k+1] \approx r_{dq}[k]$. By (4.2), $\bar{r}_{dq}$ is therefore slow varying as well, i.e. $\bar{r}_{dq}[k+1] \approx \bar{r}_{dq}[k]$.

Since the dynamics are defined in the $\alpha\beta$ reference frame, the reference vector is transformed into the same reference frame. The $\alpha\beta$ flux reference $\bar{r}_{\alpha\beta}$ is obtained using the inverse Park transformation

$$\bar{r}_{\alpha\beta} = \mathbf{T}_{dq}^{-1}(\epsilon)\bar{r}_{dq} = \begin{bmatrix} \cos\epsilon & -\sin\epsilon \\ \sin\epsilon & \cos\epsilon \end{bmatrix} \bar{r}_{dq}. \tag{4.3}$$

By Assumption 4.2, $\bar{r}_{dq}$ is slow varying with respect to T_s. However, $\bar{r}_{\alpha\beta}$ depends also on the angle ϵ. Considering Assumption 1.2, which states that the velocity ω varies slowly with respect to T_s, i.e. $\omega[k+1] \approx \omega[k]$, the angle evolves according to

$$\epsilon[k+1] = \epsilon[k] + \omega[k]T_s. \tag{4.4}$$

This relation is used to describe the flux reference dynamic in the $\alpha\beta$ reference frame

$$\begin{aligned} \bar{r}_{\alpha\beta}[k+1] &= \mathbf{T}_{dq}^{-1}(\epsilon[k] + \omega[k]T_s)\bar{r}_{dq}[k] && \text{(4.5a)} \\ &= \mathbf{T}_{dq}^{-1}(\epsilon[k] + \omega[k]T_s)\mathbf{T}_{dq}(\epsilon[k])\bar{r}_{\alpha\beta}[k] && \text{(4.5b)} \\ &= \mathbf{T}_{dq}^{-1}(\omega[k]T_s)\bar{r}_{\alpha\beta}[k], && \text{(4.5c)} \end{aligned}$$

which describes $\bar{r}_{\alpha\beta}$ as a rotating vector. It rotates with velocity ω and by the angle ωT_s at each time step T_s. Thus, $\bar{r}_{\alpha\beta}$ cannot be assumed to be slow varying in general.

Now, the reference tracking problem is transformed into a regulation problem defining the control error $x = \lambda_{\alpha\beta} - \bar{r}_{\alpha\beta}$. The state-space system (4.1) is rewritten for the state x and becomes

$$\begin{aligned} x[k+1] &= x[k] + T_s v_{\alpha\beta}[k] + \bar{r}_{\alpha\beta}[k] - \bar{r}_{\alpha\beta}[k+1] && \text{(4.6a)} \\ &= x[k] + T_s v_{\alpha\beta}[k] + (\mathbf{I} - \mathbf{T}_{dq}^{-1}(\omega[k]T_s))\bar{r}_{\alpha\beta}[k] && \text{(4.6b)} \\ &= x[k] + T_s u[k], && \text{(4.6c)} \end{aligned}$$

where the input is

$$v_{\alpha\beta}[k] = u[k] + \bar{u}[k], \tag{4.7a}$$

$$\bar{u}[k] = -\frac{1}{T_s}(\mathbf{I} - \mathbf{T}_{dq}^{-1}(\omega[k]T_s))\bar{r}_{\alpha\beta}[k]. \tag{4.7b}$$

The control input $v_{\alpha\beta}$ is the sum of a *feedforward* controller $\bar{u}$ and a *feedback* controller u. The feedforward controller $\bar{u}$ (4.7b) is necessary such that the linear error dynamics (4.6c) exists. The feedback controller u is used to achieve desired closed loop control properties, e.g. stability.

4.2. System Constraints

In Section 1.5, it is argued that state constraints, i.e. the rated current limit, do not necessarily need to be considered when controlling the states of a drive system. Roughly speaking, the state constraints are inherently "soft" such that short term violations can be accepted. It is sufficient that the state converges to a reference value, which satisfies the state constraints. On the other hand, the inverter provides only a finite voltage and the input limits need to be taken into account. In this chapter, the input $v_{\alpha\beta}$ is applied via modulation (SVM/PWM) to the inverter[1]. These $v_{\alpha\beta}$ are confined by the *convex control set* $\mathcal{V}_d$ (3.9). On some occasions, it is convenient to restrict this set to the convex $\mathcal{V} \subset \mathcal{V}_d$ (3.10), to simplify the treatment. Thus, the input is required to satisfy the following equivalent requirements

$$v_{\alpha\beta} \in \mathcal{V}_d \quad \Leftrightarrow \quad u + \bar{u} \in \mathcal{V}_d \quad \Leftrightarrow \quad u \in \mathcal{U}_d. \tag{4.8}$$

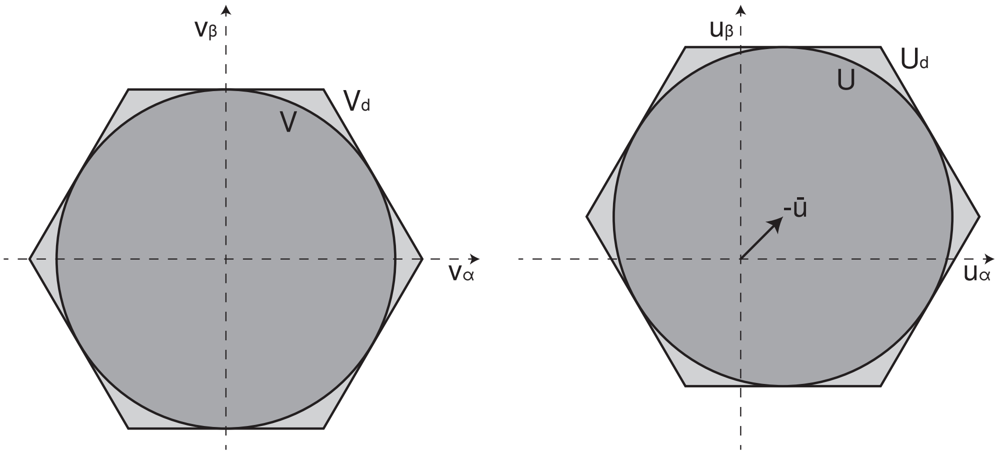
The set $\mathcal{U}_d$ is understood to be the $-\bar{u}$ displaced set $\mathcal{V}_d$ and it is defined by

$$\mathcal{U}_d \overset{\text{def}}{=} \mathcal{V}_d - \bar{u} = \left\{ u \in \mathbb{R}^2 \mid \mathbf{H}u \leq v_r - \mathbf{H}\bar{u} \right\}, \tag{4.9}$$

and the matrix $\mathbf{H}$ is

$$\mathbf{H} \overset{\text{def}}{=} \begin{bmatrix} 0 & 1 \\ \sqrt{3}/2 & 1/2 \\ \sqrt{3}/2 & -1/2 \\ 0 & -1 \\ -\sqrt{3}/2 & -1/2 \\ -\sqrt{3}/2 & 1/2 \end{bmatrix}. \tag{4.10}$$

[1] Applying the input via modulation is opposed to direct actuation, which produces the *finite control set* $\mathcal{V}_s$ (3.6). This set is an integer set and therefore nonconvex.



(a) Set $\mathcal{V}_d$, i.e. $\mathcal{V}$

(b) Set $\mathcal{U}_d$, i.e. $\mathcal{U}$

Figure 4.1.: Relation of the sets $\mathcal{V}_d$ and $\mathcal{U}_d$, i.e. $\mathcal{V}$ and $\mathcal{U}$

The l-th row of $\mathbf{H}$ is denoted with $\mathbf{H}_l$. Moreover, $\mathcal{U}$ denotes the $-\bar{u}$ displaced set $\mathcal{V}$, which is defined as

$$\mathcal{U} \overset{\text{def}}{=} \mathcal{V} - \bar{u} = \Big\{ u \in \mathbb{R}^2 \mid \|u + \bar{u}\| \leq v_r \Big\} \subset \mathcal{U}_d. \tag{4.11}$$

The relation of the sets $\mathcal{V}_d$ and $\mathcal{U}_d$, i.e. $\mathcal{V}$ and $\mathcal{U}$, is shown graphically in Figure 4.1. The purpose of the input u is to implement a stabilizing feedback. A helpful property is that $\mathcal{U}_d$, i.e. $\mathcal{U}$, contains an arbitrarily small neighborhood of the origin. Formally, the set $\mathcal{U} \subset \mathcal{U}_d$ is required to contain an arbitrarily small ball

$$\mathcal{B} \overset{\text{def}}{=} \Big\{ u \in \mathbb{R}^2 \mid \|u\| \leq b \Big\} \subset \mathcal{U} = \mathcal{V} - \bar{u}, \tag{4.12}$$

where $b \in \mathbb{R}_{>0}$ is an arbitrarily small positive integer. Considering the system dynamics, this requirement can be interpreted as prerequisite such that $u \in \mathcal{U} \subset \mathcal{U}_d$ can steer the state in any direction.

The set $\mathcal{V}$ is nonempty, i.e. $\mathcal{V} \neq \emptyset$, since the DC link voltage is assumed to be positive throughout this text. Moreover, it is a two-dimensional ball centered in the origin and the statement $\mathcal{B} \subset \mathcal{V}$ is always true for a sufficiently small $b > 0$. Consequently, (4.12) translates into a condition on the feedforward controller $\bar{u}$. Clearly, (4.12) is conservative if the constraint $u \in \mathcal{U}_d$ is used since $\mathcal{U} \subset \mathcal{U}_d$. In fact, the condition could be relaxed to $\mathcal{B} \subset \mathcal{U}_d$ but is more complex to treat. Moreover, $\mathcal{B} \subset \mathcal{U}$ does not significantly deteriorate the system behavior such that (4.12) is suitable as a general constraint.

Proposition 4.1. *Let $\bar{u}$ belong to the interior of $\mathcal{V}$, i.e. $\bar{u} \in \operatorname{int} \mathcal{V}$, then (4.12) holds.*

Proof. The set $\mathcal{V} \neq \emptyset$ is convex and $\mathcal{B} \subset \mathcal{V}$ for a sufficiently small $b > 0$. Thus, any $\bar{u} \in \operatorname{int} \mathcal{V}$ can be subtracted from the set such that the remaining set $\mathcal{U} = \mathcal{V} - \bar{v}$ still contains a sufficiently small ball $\mathcal{B}$ centered in the origin. □

Now, the impact of the condition $\bar{u} \in \operatorname{int} \mathcal{V}$ on the operation of the system is analyzed. The vector $\bar{u}$ is a function of the reference $\bar{r}_{\alpha\beta}$ and depends on the system parameter ω. Thus, a simple criterion on $\bar{r}_{\alpha\beta}$, i.e. $\bar{r}_{dq}$, is derived such that $\bar{u} \in \operatorname{int} \mathcal{V}$ holds. It is generally required that the reference $\bar{r}_{dq}$ satisfies the constraints defined in Section 1.5 and ideally they are sufficient such that $\bar{u} \in \operatorname{int} \mathcal{V}$. In this context, the constraint (1.33) is relevant and provides the following equivalent statements

$$\omega \bar{r}_{dq} \in \bar{\mathcal{V}} \quad \Leftrightarrow \quad |\omega| \|\bar{r}_{dq}\| \leq \rho_v v_r \quad \Leftrightarrow \quad |\omega| \|\bar{r}_{\alpha\beta}\| \leq \rho_v v_r \tag{4.13}$$

with $\rho_v \in (0,1)$ and $\|\bar{r}_{\alpha\beta}\| = \|\mathbf{T}_{dq}^{-1}(\epsilon)\bar{r}_{dq}\| = \|\bar{r}_{dq}\|$ since $\mathbf{T}_{dq}^{-1}(\epsilon)$ is orthogonal. Using these properties, the following can be said.

Proposition 4.2. *Let $|\omega| \|\bar{r}_{dq}\| = |\omega| \|\bar{r}_{\alpha\beta}\| \leq \rho_v v_r$ with $\rho_v \in (0,1)$, then $\bar{u} \in \operatorname{int} \mathcal{V}$.*

Proof. First, it is shown that $\|\bar{u}\|$ is upper bounded by $|\omega| \|\bar{r}_{\alpha\beta}\|$, since

$$\begin{aligned}
\|\bar{u}\| &\leq |\omega| \|\bar{r}_{\alpha\beta}\|, && \text{(4.14a)}\\
\left\|(\mathbf{I} - \mathbf{T}_{dq}^{-1}(\omega T_s))\bar{r}_{\alpha\beta}\right\| &\leq |\omega| T_s \|\bar{r}_{\alpha\beta}\|, && \text{(4.14b)}\\
\sqrt{2}\sqrt{1 - \cos \omega T_s} \|\bar{r}_{\alpha\beta}\| &\leq |\omega| T_s \left\|\bar{r}_{\alpha\beta}\right\|, && \text{(4.14c)}\\
1 - \frac{\omega^2 T_s^2}{2} &\leq \cos \omega T_s, && \text{(4.14d)}
\end{aligned}$$

is always true. Thus, $|\omega| \|\bar{r}_{dq}\| = |\omega| \|\bar{r}_{\alpha\beta}\| \leq \rho_v v_r$ implies that

$$\|\bar{u}\| \leq |\omega| \|\bar{r}_{\alpha\beta}\| \leq \rho_v v_r < v_r, \tag{4.15}$$

for all $\rho_v \in (0,1)$. Therefore, $\bar{u} \in \operatorname{int} \mathcal{V}$ holds by the definition (3.10). □

Remark 4.1. Any dq current reference r_{dq} obtained with the reference generation procedure of Section 2.4, which is transformed into a dq flux reference with $\bar{r}_{dq} = \mathbf{L} r_{dq} + \psi_{dq}$ and into a $\alpha\beta$ flux reference with $\bar{r}_{\alpha\beta} = \mathbf{T}_{dq}^{-1}(\epsilon)\bar{r}_{dq}$, yields a feedforward controller $\bar{u}$, that satisfies $\bar{u} \in \operatorname{int} \mathcal{V}$.

4.3. Constrained Feedback Control

In this section, the nonlinear feedback controller is introduced. Clearly, there does *not* exists a linear feedback controller $u = \kappa_l x$ which produces an admissible control action $\kappa_l x \in \mathcal{U}/\mathcal{U}_d$[2] for all $x \in \mathbb{R}^2$ due to the presence of system (input) constraints. Thus, a nonlinear controller $u = \kappa(x)$ is designed, which is globally *feasible*, i.e. $\kappa(x) \in \mathcal{U}/\mathcal{U}_d$ for all $x \in \mathbb{R}^2$, and results in a globally *stable* closed loop system. If such a $\kappa(x)$ exists, this implies that (i) there exists an admissible $u \in \mathcal{U}/\mathcal{U}_d$ for all $x \in \mathbb{R}^2$, and (ii) the system is globally *stabilizable*, i.e. there exists an admissible $u \in \mathcal{U}/\mathcal{U}_d$ such that a global control Lyapunov function is decreasing [3], [91].

The nonlinear feedback controller $\kappa(x)$ is defined based on a simple linear controller for the unconstrained system (4.6c)

$$\tilde{u} = -\frac{K}{T_s}x, \tag{4.16}$$

where $K \in (0, 2) \subset \mathbb{R}$ is the tuning parameter. To take the system (input) constraints into account, a nonlinear parameter $\xi(x) \in (0, 1] \subset \mathbb{R}$ is added, which yields the nonlinear feedback control law

$$u = \kappa(x) \overset{\text{def}}{=} \xi(x)\tilde{u} = -\xi(x)\frac{K}{T_s}x. \tag{4.17}$$

4.4. Feasibility and Stability

Applying the nonlinear controller (4.17) to (4.6c) leads to a closed loop system with well defined properties. The feasibility, i.e. the existence, of the controller $\kappa(x) \in \mathcal{U}/\mathcal{U}_d$ is shown as follows.

Proposition 4.3. *If $\bar{u} \in \operatorname{int}\mathcal{V}$, then the nonlinear feedback controller*

$$\kappa(x) = \xi(x)\tilde{u} = -\xi(x)\frac{K}{T_s}x \in \mathcal{U}/\mathcal{U}_d, \tag{4.18}$$

exists for all $x \in \mathbb{R}^2$ by choosing a sufficiently small $\xi(x) \in (0, 1]$.

Proof. Since x (and thus $\tilde{u}$) is a vector with finite magnitude, there exists a sufficiently small $\xi(x) \in (0, 1]$ such that $\kappa(x) = \xi(x)\tilde{u}$ is in the neighborhood of the origin, i.e. $\kappa(x) \in \mathcal{B}$ for an arbitrary $b > 0$. If $\bar{u} \in \operatorname{int}\mathcal{V}$ (and $b > 0$ is sufficiently small), then $\mathcal{B} \subset \mathcal{U}/\mathcal{U}_d$ by Proposition 4.1 and thus $\kappa(x) \in \mathcal{U}/\mathcal{U}_d$. □

[2] The notation $\mathcal{U}/\mathcal{U}_d$ is introduced to denote either $\mathcal{U}$ or $\mathcal{U}_d$. In other words, the statement $\mathcal{U}/\mathcal{U}_d$ can be replaced by $\mathcal{U}$ or $\mathcal{U}_d$ dependent on the preference.

Thus, (4.17) is an admissible control law for the *constrained* system. The stability of the closed loop system is shown as follows.

Theorem 4.1. *If $\bar{u} \in \operatorname{int} \mathcal{V}$, then the closed loop system (4.6c) with feedback control law (4.17) is asymptotically stable for all $\xi(x) \in (0, 1]$ and $K \in (0, 2)$.*

Proof. If $\bar{u} \in \operatorname{int} \mathcal{V}$, then the feedback controller $u = \kappa(x)$ exists by Proposition 4.3. Thus, the closed loop system is

$$x[k+1] = x[k] + T_s \kappa(x[k]) = (1 - \xi(x)K)x, \tag{4.19}$$

where $(1 - \xi(x)K) \in (-1, 1)$ for all $\xi(x) \in (0, 1]$ and $K \in (0, 2)$.

Let $\Gamma(x) \overset{\text{def}}{=} \|x\|_g$ with $g \in \{1, 2, \infty\}$ be a global candidate Lyapunov function ($\Gamma(x) > 0$ for all $x \neq 0$; $\Gamma(0) = 0$; $\Gamma(x)$ is radially unbounded and continuous in the origin). The subscript g with $g \in \{1, \infty\}$ denotes the 1 and ∞ norm $\|x\|_1$ and $\|x\|_\infty$, respectively. Moreover, $g = 2$ denotes the squared Euclidean norm $\|x\|_2 = (\|x\|)^2$ for notational simplicity.

If $\xi(x)K \in (0, 1]$, i.e. $(1 - \xi(x)K) \in [0, 1)$, then

$$\Gamma(x + T_s \kappa(x)) = \|(1 - \xi(x)K)x\|_g = (1 - \xi(x)K)\|x\|_g.$$

Therefore, the candidate Lyapunov function difference is

$$\Delta\Gamma(x) = \Gamma(x + T_s \kappa(x)) - \Gamma(x) = -\xi(x)K\|x\|_g. \tag{4.20}$$

Since $-\xi(x)K < 0$ for all $\xi(x)K \in (0, 1]$, the Lyapunov difference equation $\Delta\Gamma(x) < 0 \ \forall x \neq 0$ and $\Delta\Gamma(0) = 0$, i.e. $\Gamma(x)$ is a Lyapunov function and the system is asymptotically stable.

If $\xi(x)K \in [1, 2)$, i.e. $(1 - \xi(x)K) \in (-1, 0]$, then

$$\Gamma(x + T_s \kappa(x)) = \|(1 - \xi(x)K)x\|_g = (\xi(x)K - 1)\|x\|_g.$$

Therefore, the candidate Lyapunov function difference is

$$\Delta\Gamma(x) = \Gamma(x + T_s \kappa(x)) - \Gamma(x) = (\xi(x)K - 2)K\|x\|_g. \tag{4.21}$$

Since $(\xi(x)K - 2) < 0$ for all $\xi(x)K \in [1, 2)$, the Lyapunov difference equation $\Delta\Gamma(x) < 0 \ \forall x \neq 0$ and $\Delta\Gamma(0) = 0$, i.e. $\Gamma(x)$ is a Lyapunov function and the system is asymptotically stable. □

4.5. Implementation

In this section, the implementation of the controller $\kappa(x)$ is discussed. Since $\tilde{u} = -\frac{K}{T_s}x$ is known when computing the controller, the confinement parameter $\xi(x)$ has to be chosen such that $\kappa(x) = \xi(x)\tilde{u} \in \mathcal{U}/\mathcal{U}_d$. The conditions such that the feedback controller $\kappa(x) \in \mathcal{U}/\mathcal{U}_d$ exists are defined by Proposition 4.3. It requires that the parameter $\xi(x)$ is sufficiently small such that $\kappa(x) \in \mathcal{U}/\mathcal{U}_d$. On the other hand, $\xi(x)$ should be large ($\xi(x) \to 1$) to minimize the control with respect to the confinement parameter. In other words, the controller $\kappa(x)$ should be as close as possible to the unconstrained controller $\tilde{u}$ but should respect the input constraints. Thus, $\xi(x)$ is defined as the solution of the optimization problem

$$\xi(x) \overset{\text{def}}{=} \underset{\xi\in(0,1]}{\text{maximize}}\ \xi \tag{4.22a}$$

$$\text{subject to } \xi\tilde{u} \in \mathcal{U}/\mathcal{U}_d; \tag{4.22b}$$

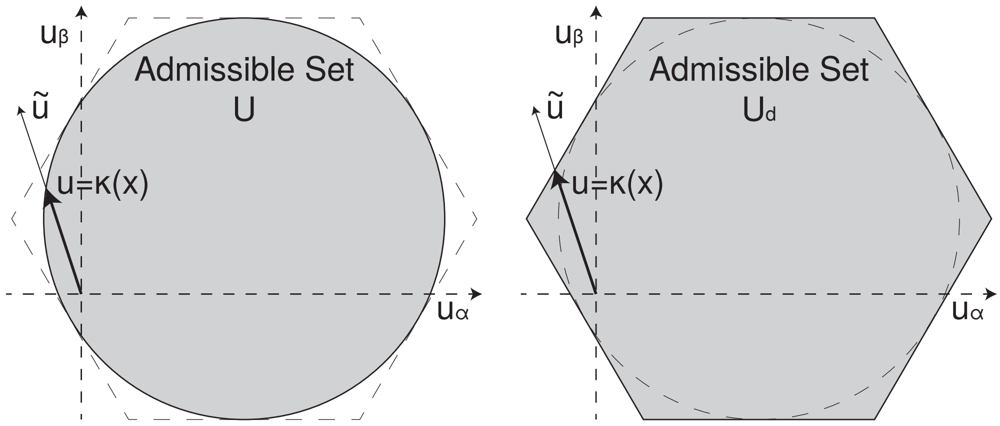
which is solved by the following algorithms. The solver policy is slightly different for $\kappa(x) \in \mathcal{U}$ and $\kappa(x) \in \mathcal{U}_d$ and leads to Algorithm 4.1 and Algorithm 4.2, respectively. The optimization problem (4.22) is shown graphically in Figure 4.2.

The optimization problem (4.22) using the constraint $\xi\tilde{u} \in \mathcal{U}$ is shown in Figure 4.2(a). Initially, it is assumed that $\xi = 1$, which is the solution to (4.22) if $\xi\tilde{u} = \tilde{u} \in \mathcal{U}$. Otherwise, the solution (maximum ξ) lies on the border of the set $\mathcal{U}$, i.e.

$$\xi\tilde{u} \in \text{br}\,\mathcal{U} \quad \Leftrightarrow \quad \|\xi\tilde{u} + \bar{u}\| = v_r \quad \Leftrightarrow \quad \xi^2\tilde{u}'\tilde{u} + \xi 2\tilde{u}'\bar{u} + \bar{u}'\bar{u} - v_r^2 = 0. \tag{4.23}$$

Algorithm 4.1: Algorithm to solve (4.22) for $\kappa(x) \in \mathcal{U}$

```
if $\|\bar{u}\| < v_r$ then
    /* $\bar{u} \in \text{int}\,\mathcal{V}$; $\kappa(x) \in \mathcal{U}$ exists */
    $\xi \leftarrow 1$    /* initialize optimization variable */
    if $\|\bar{u} + \tilde{u}\| > v_r$ then
        $\xi \leftarrow \frac{-\tilde{u}'\bar{u}+\sqrt{(\tilde{u}'\bar{u})^2+(\tilde{u}'\tilde{u})(v_r-\bar{u}'\bar{u})}}{\tilde{u}'\tilde{u}}$    /* $\bar{u} + \xi\tilde{u} \notin \mathcal{V}$; limit $\xi$ */
    $\xi(x) \leftarrow \xi$    /* assign optimal result */
else
    /* $\bar{u} \notin \text{int}\,\mathcal{V}$; $\kappa(x) \in \mathcal{U}$ does not exist in general */
    stop
```



(a) Using $\kappa(x) \in \mathcal{U}$

(b) Using $\kappa(x) \in \mathcal{U}_d$

Figure 4.2.: Optimization problem (4.22)

If $\bar{u} \in \operatorname{int} \mathcal{V}$ then $(v_r^2 - \bar{u}'\bar{u}) > 0$, which yields always two real solutions to the quadratic equation. The positive solution

$$\xi = \frac{-\tilde{u}'\bar{u} + \sqrt{(\tilde{u}'\bar{u})^2 + (\tilde{u}'\tilde{u})(v_r^2 - \bar{u}'\bar{u})}}{\tilde{u}'\tilde{u}} > 0, \tag{4.24}$$

solves (4.22) with $\xi(x) = \xi$.

The optimization problem (4.22) using the constraint $\xi\tilde{u} \in \mathcal{U}_d$ is shown in Figure 4.2(b). The solution procedure is similar to the previous case starting with the

Algorithm 4.2: Algorithm to solve (4.22) for $\kappa(x) \in \mathcal{U}_d$

```
if $\|\bar{u}\| < v_r$ then
    /* $\bar{u} \in \operatorname{int}\mathcal{V}$; $\kappa(x) \in \mathcal{U}_d$ exists */
    $\xi \leftarrow 1$                    /* initialize optimization variable */
    for $l \leftarrow 1$ to 6 do
        /* for all rows of $\mathbf{H}$ do */
        if $\xi\mathbf{H}_l\tilde{u} > v_r - \mathbf{H}_l\bar{u}$ then
            $\xi \leftarrow \frac{v_r - \mathbf{H}_l\bar{u}}{\mathbf{H}_l\tilde{u}}$     /* $l$-th row not satisfied by $\xi\tilde{u}$; limit $\xi$ */
    $\xi(x) \leftarrow \xi$                          /* assign optimal result */
else
    /* $\bar{u} \notin \operatorname{int}\mathcal{V}$; $\kappa(x) \in \mathcal{U}_d$ does not exist in general */
    stop
```

assumption $\xi = 1$, which is then checked and eventually corrected. However, $\mathcal{U}_d$ is a polytope defined by the intersection of halfplanes. Each halfplane is defined by the rows $\mathbf{H}_l$ of $\mathbf{H}$ in (4.9). Therefore, each of the $l = [1, 6]$ linear constraints is checked separately. If ξ violates the l-th constraint, i.e.

$$\xi \mathbf{H}_l \tilde{u} > v_r - \mathbf{H}_l \bar{u}, \tag{4.25}$$

the vector $\xi \tilde{u}$ needs to lie on the border of that halfplane and ξ is reduced accordingly with

$$\xi = \frac{v_r - \mathbf{H}_l \bar{u}}{\mathbf{H}_l \tilde{u}}. \tag{4.26}$$

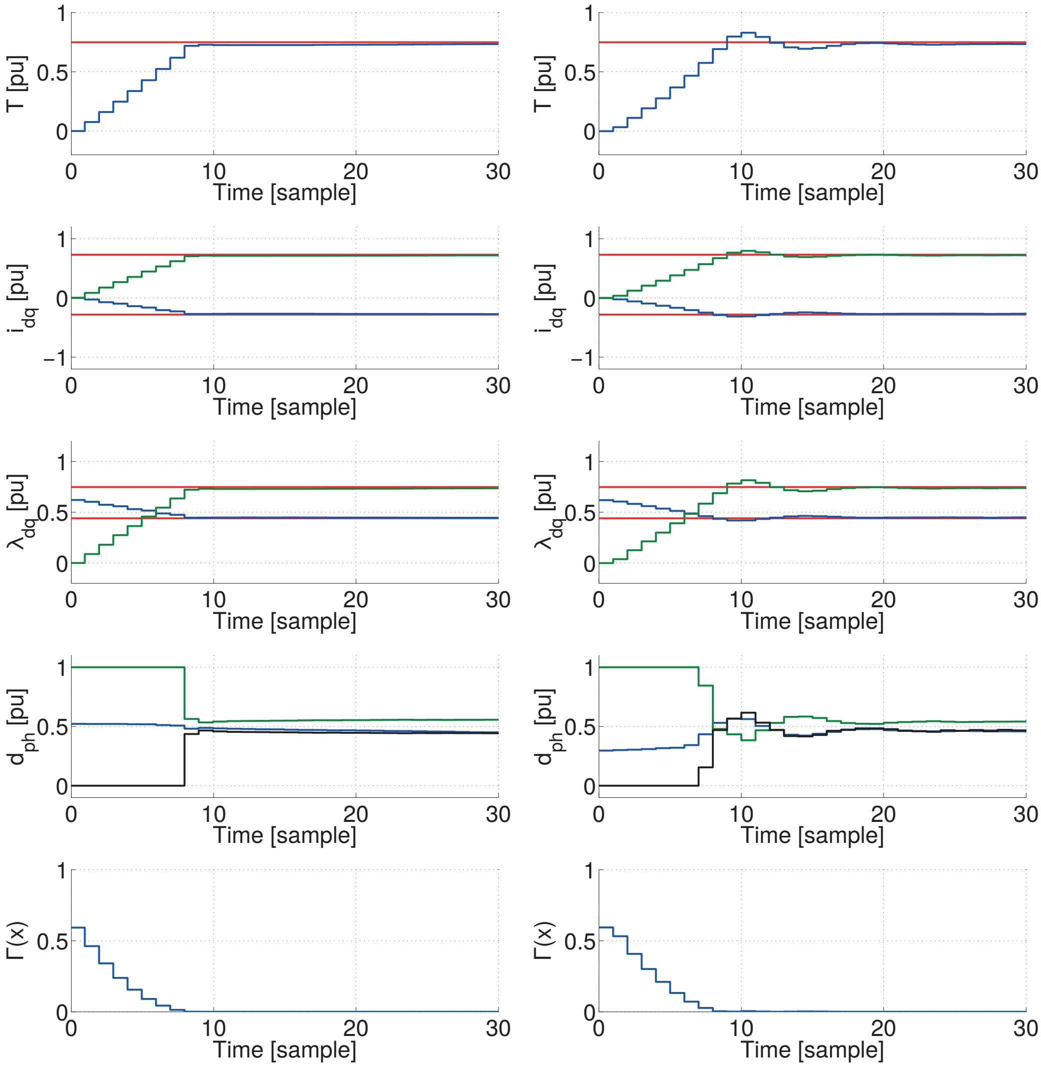
When all 6 rows of the matrix $\mathbf{H}$ have been checked, $\xi \tilde{u}$ is known to satisfy each of the halfplane constraints and therefore $\xi \tilde{u} \in \mathcal{U}_d$. Thus, it is the solution to (4.22) with $\xi(x) = \xi$.

In real-time, the execution time of the control code is critical due to the small sampling times of power electronic systems and the limited computation capabilities of embedded control hardware. Thus, the Algorithm 4.1 and Algorithm 4.2 are benchmarked on the simulation platform and the experimental test bench, which are outlined in Appendix D. The results are shown in Table 5.1 in Chapter 5. Both algorithms are not particularly demanding in terms of computation requirements and are comparable with e.g. anti-windup proportional-integral control. Thus, the resulting execution times are used as benchmark references for the predictive control implementations in the next chapters.

4.6. Dynamic Operation

In this section, the proposed nonlinear controller $\kappa(x) \in \mathcal{U}_d$ with $K = 1$ and optimal reference generation procedure is reviewed in dynamic operation. First, a torque reference step is applied to the reference generation procedure. This results in both a d and q current, i.e. flux, reference step. The step is applied at standstill $\omega = 0$ and during the period of observation, the machine speed remains approximately zero. The simulation and experimental results are shown in Figure 4.3. From top, the figure shows the reference (R, red) and actual (T, blue) torque; the reference current vector (r_{dq}, red), the actual d (i_d, blue) and q (i_q, green) current; the reference flux vector ($\bar{r}_{dq}$, red), the actual d (λ_d, blue) and q (λ_q, green) flux; the duty cycle d_{ph}; and the Lyapunov function $\Gamma(x)$ with $x = \lambda_{\alpha\beta} - \bar{r}_{\alpha\beta}$.

The results confirm the nonlinear control design and implementation. They show that the Lyapunov function is strictly decreasing once the reference step is applied.

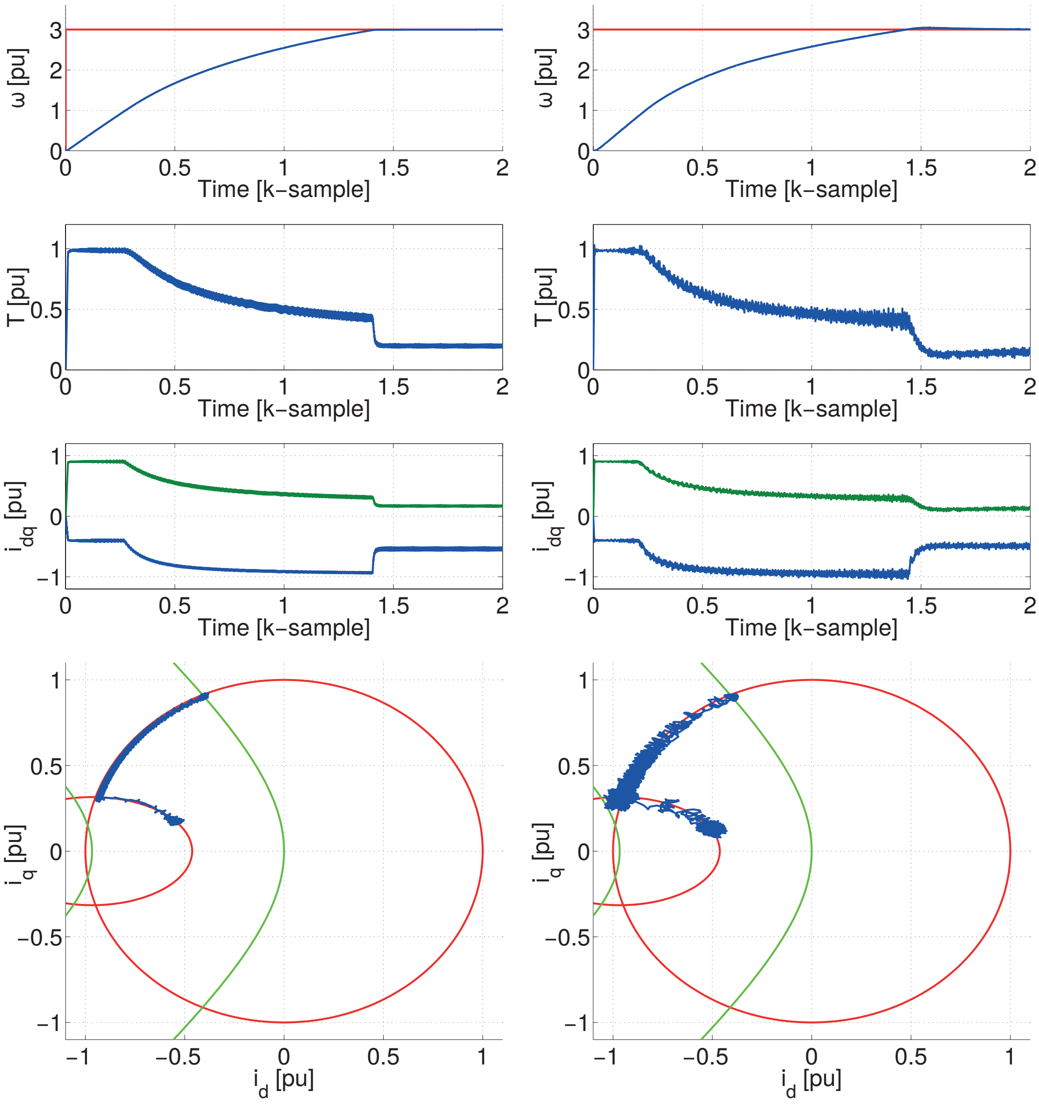


(a) Simulation

(b) Experimentation

Figure 4.3.: Torque step from $0pu$ to $0.75pu$ using the nonlinear controller $\kappa(x)$

The controller applies the maximum voltage according to the input constraint $\mathcal{U}_d$ until the reference is reached. As a consequence the raise time depends on the electric angle, which defines the voltage magnitude in a certain direction. However, the raise time is at least as good as using the input constraint $\mathcal{U}$, where the voltage magnitude is constant. Comparing simulation and experimentation, an overshoot is observed on the latter results. This effect is introduced by the observer, which is



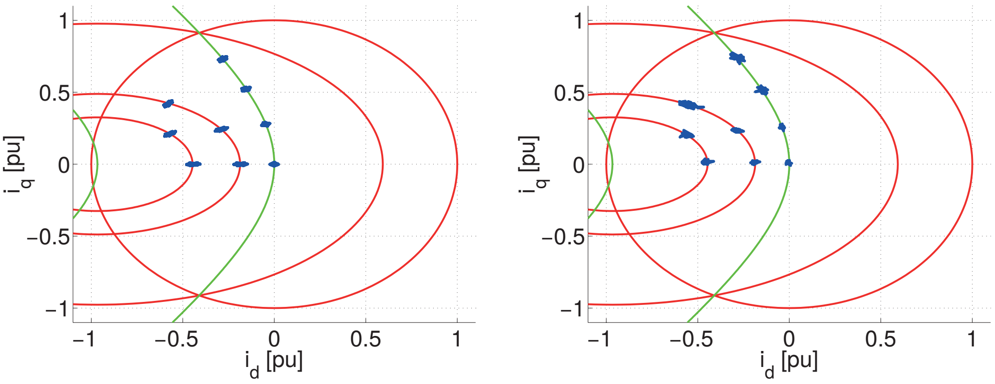
(a) Simulation

(b) Experimentation

Figure 4.4.: Speed step form standstill to $3pu$ using the nonlinear controller $\kappa(x)$

treated in Appendix C. The observer converges slower than the controller and relies on integration in presence of model uncertainties.

Moreover, a speed reference step from standstill to $3pu$ is evaluated using an anti-windup proportional-integral speed controller. Since the reference step is large, the



(a) Simulation

(b) Experimentation

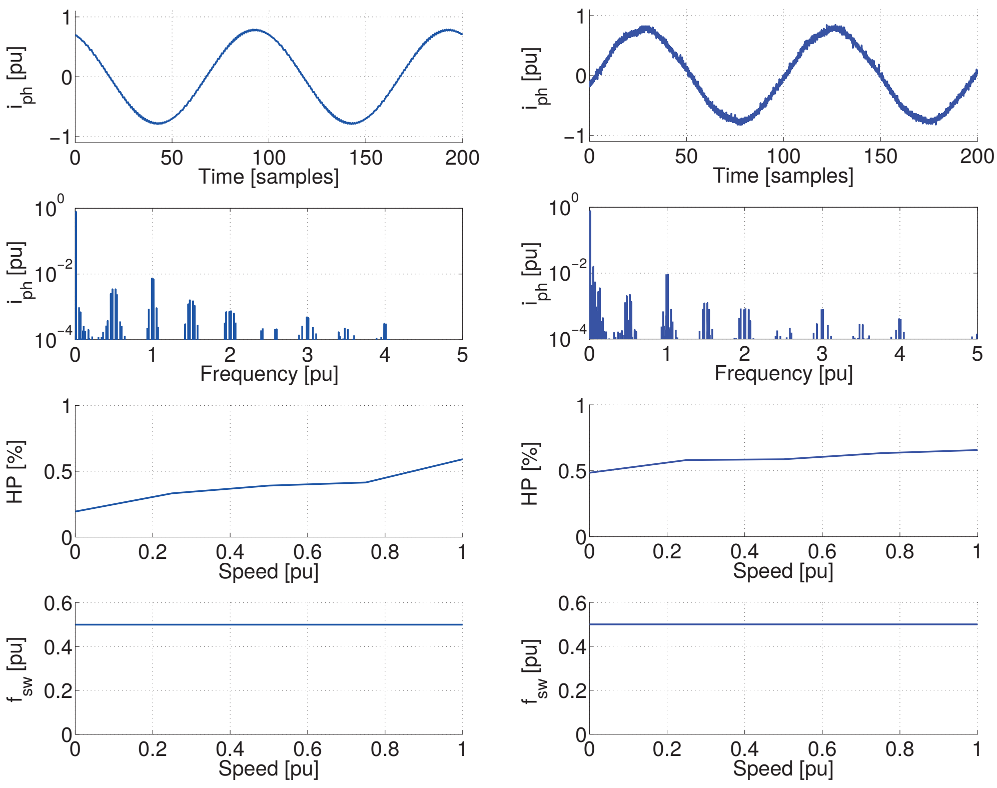
Figure 4.5.: Steady-state operation points using the nonlinear controller $\kappa(x)$

speed loop saturates and the maximum available torque is applied approximately until the speed reference is reached. The results are presented in Figure 4.4. From top, the figure shows the reference (red) and actual (ω, blue) speed; the torque (T); the actual d (i_d, blue) and q (i_q, green) current; and the current i_{dq} in the current state-plane.

The results confirm the design and implementation of the nonlinear controller and the reference generation procedure. At low speed, the maximum torque and acceleration are constant. The current producing maximum torque lies on the intersection of the MTPA and the isocurrent locus. Increasing the speed beyond rated speed, this operation point is not available anymore due to the voltage limit. Thus, the current moves along the isocurrent locus until the reference speed is achieved. Then the current moves along the isoflux locus to the operation point which provides the torque to maintain the machine at reference speed.

4.7. Steady-State Operation

In this section, the proposed nonlinear controller $\kappa(x) \in \mathcal{U}_d$ with $K = 1$ and optimal reference generation procedure is reviewed in steady-state operation. First, different load torques are applied such that the machine provides a constant torque T at different speeds ω. The cases $\omega = 1.0pu$ with $T = \{0pu, 0.25pu, 0.5pu, 0.75pu\}$; $\omega = 2.0pu$ with $T = \{0pu, 0.25pu, 0.5pu\}$; and $\omega = 3.0pu$ with $T = \{0pu, 0.25pu\}$; are evaluated. The simulation and experimental results are shown in Figure 4.5. Ideally, steady-state operation with given conditions (torque, speed) would lead to a single



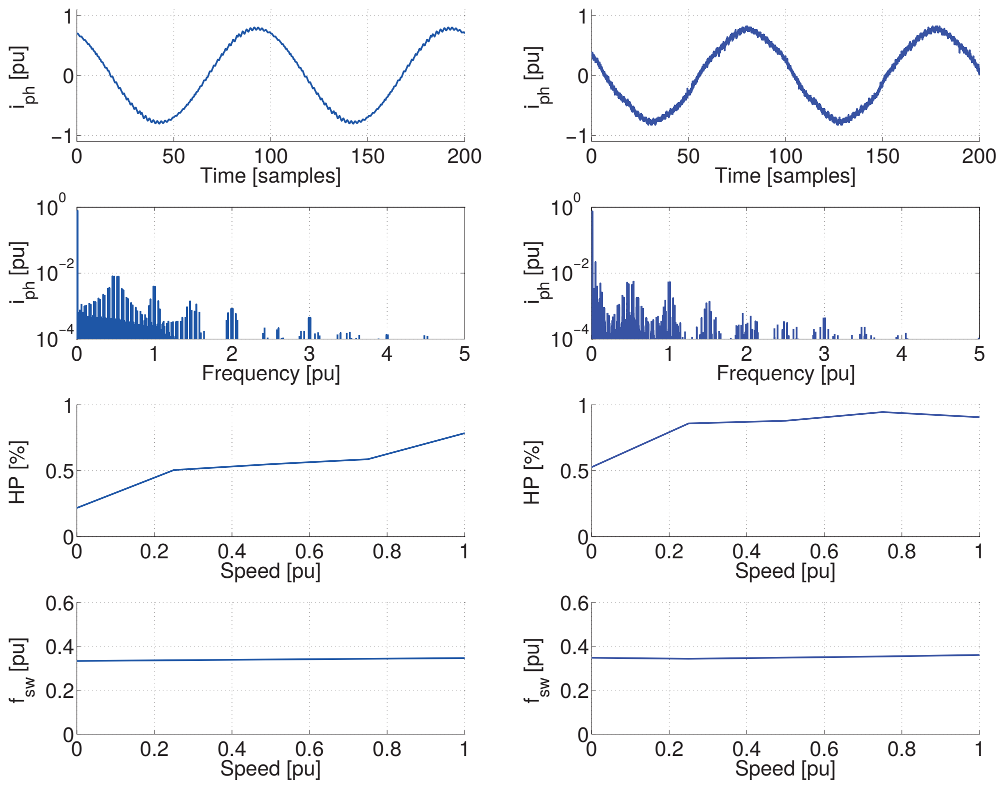
(a) Simulation

(b) Experimentation

Figure 4.6.: Current quality using the nonlinear controller $\kappa(x)$ and SSVM

point on the state plane. In practice, the states stay in the neighborhood of this point due to power converter switching, measurement noise, etc. The results show that the drive system applies states according to the reference generation procedure. The controller operates on the MTPA trajectory at $\omega = 1.0pu$. At $\omega = 2.0pu$ and $\omega = 3.0pu$, the system operates on the isoflux locus, which is defined by the speed ω.

The evaluation is completed highlighting the steady-state current quality, which is achieved by the nonlinear controller with $K = 1$. Using a modulator, the switching related harmonics depend mainly on the underlying modulation scheme. The controller is combined and evaluated with the symmetric SVM (SSVM), which is shown in Figure 4.6, and the discontinuous SVM (DSVM), which is depicted in Figure 4.7. From top, the typical current waveform and spectrum is presented (at $0.75pu$ speed and $0.75pu$ torque). In these figures, frequencies are normalized with



(a) Simulation

(b) Experimentation

Figure 4.7.: Current quality using the nonlinear controller $\kappa(x)$ and DSVM

the sampling frequency $f_s = 1/T_s$. Moreover, the harmonic power (HP) and the average switching frequency f_{sw} of the three legs are evaluated as a function of the machine speed.

The current harmonics of a drive system have multiple origins. Low frequency harmonics of a drive system are introduced by the electrical machine, e.g. due to harmonics in the back-EMF. In this section, the focus lies on harmonics, which are introduced by control and switching. Thus, the performance index *Harmonic Power* is introduced

$$HP \overset{\text{def}}{=} \sqrt{\frac{1}{3}\int_{0.1f_s}^{10f_s} i'_{ph}(f) i_{ph}(f) df}. \tag{4.27}$$

It captures the harmonics of interest but disregards low frequency machine specific harmonics evaluating the frequency band from $0.1f_s$ to $10f_s$. The index is proportional to the power of the harmonics in the defined harmonic band as average of the three phases.

The switching frequency is evaluated counting the number of switch transitions per phase $n_{ph} \in \mathbb{N}^3$ in an evaluation interval $T_{sw} \in \mathbb{R}_+$. Thus, the switching frequency per leg can be computed with

$$f_{sw,ph} = \frac{1}{2}\frac{1}{T_{sw}} n_{ph} \in \mathbb{R}^3_+, \tag{4.28}$$

and the mean switching frequency of the inverter [84]

$$f_{sw} \overset{\text{def}}{=} \frac{1}{2}\frac{1}{3T_{sw}} 1' n_{ph} \in \mathbb{R}_+. \tag{4.29}$$

The factor 1/2 is required if both the turn-*on* and turn-*off* switching transitions are counted.

Both SSVM and DSVM exhibit significant switching harmonics in the neighborhood of the PWM carrier frequency [65] and its multiples. The carrier frequency is half the sampling frequency since the drive system is sampled at the positive and negative peak of the triangular carrier frequency. In SSVM, the carrier frequency corresponds to the average switching frequency leading to distinct switching harmonics and low noise. In DSVM, the switching frequency is reduced to approximately 2/3 of the carrier frequency decreasing the current quality, i.e. increasing the HP. Both modulation concepts achieve a constant switching frequency, which is independent of the machine speed[3]. Comparing the simulation and experimental results, the current quality deteriorates to some extend. The main causes are measurement noise and the presence of machine dependent low order harmonics (mainly back-emf), which are not considered in simulation.

[3] Using DSVM, two legs are switched instead of three in each sampling period. However, there are sampling instants where all three legs are switched. The number of such instants is proportional to the fundamental frequency of the current such that the switching frequency is only approximately independent from the machine speed.

Chapter 5.

Convex Control Set (CCS) MPC

In this chapter, Convex Control Set (CCS) Model Predictive Control (MPC) is designed in the virtual flux space. CCS specifies the input constrained set to be convex, which is obtained using a modulation scheme (PWM/SVM). Stability is achieved using the Lyapunov-based MPC approach, where neither requirements on the prediction horizon nor a terminal set is needed. Efficient algorithms are presented such that the CCS-MPC with prediction horizon equal to one can be implemented on embedded control hardware meeting typical real time requirements of power electronic systems. Longer prediction horizons can be achieved using online numerical or offline parametric solvers. The CCS-MPC design is evaluated in simulation and on an experimental test bench.

5.1. System

In this section, the system dynamics and constraints are defined. Since the system is the same as the one defined in Chapter 4, only a summary is provided here. The model is based on the discrete-time state-space formulation in the $\alpha\beta$ reference frame (1.19)

$$\lambda_{\alpha\beta}[k+1] = \lambda_{\alpha\beta}[k] + T_s v_{\alpha\beta}[k], \tag{5.1}$$

assuming that disturbances (resistive voltage drop, inverter nonlinear effects, etc.) are compensated externally. The control goal is to track an $\alpha\beta$ reference vector with the dynamics

$$\bar{r}_{\alpha\beta}[k+1] = \mathbf{T}_{dq}^{-1}(\omega[k]T_s)\bar{r}_{\alpha\beta}[k]. \tag{5.2}$$

In other words, $\bar{r}_{\alpha\beta}$ rotates with velocity ω and by the angle ωT_s at each time step T_s. The reference tracking problem is transformed into a regulation problem using the control error $x = \lambda_{\alpha\beta} - \bar{r}_{\alpha\beta}$ as state x. This approach yields the dynamic model

$$\begin{aligned} x[k+1] &= x[k] + T_s v_{\alpha\beta}[k] - T_s \bar{u}[k] && (5.3a) \\ &= x[k] + T_s u[k], && (5.3b) \end{aligned}$$

with the input

$$\begin{aligned} u[k] &= v_{\alpha\beta}[k] - \bar{u}[k], && (5.4a) \\ \bar{u}[k] &= -\frac{1}{T_s}(\mathbf{I} - \mathbf{T}_{dq}^{-1}(\omega[k]T_s))\bar{r}_{\alpha\beta}[k]. && (5.4b) \end{aligned}$$

The original control input $v_{\alpha\beta}$ is the sum of a *feedforward* controller $\bar{u}$ and a *feedback* controller u. The feedforward controller $\bar{u}$ (5.4b) is necessary such that the linear error dynamics (5.3b) exists. The feedback controller u is used to achieve closed loop control properties.

The input $v_{\alpha\beta}$ is applied via modulation scheme (SVM/PWM). Thus, the valid inputs $v_{\alpha\beta}$ are confined by the *convex control set* $\mathcal{V}_d$ (3.9) or the smaller $\mathcal{V} \subset \mathcal{V}_d$ (3.10). The input is required to satisfy the requirement

$$v_{\alpha\beta} \in \mathcal{V}_d \quad \Leftrightarrow \quad u + \bar{u} \in \mathcal{V}_d \quad \Leftrightarrow \quad u \in \mathcal{U}_d. \tag{5.5}$$

The set $\mathcal{U}_d$, i.e. $\mathcal{U}$, is defined to be the $-\bar{u}$ displaced set $\mathcal{V}_d$, i.e. $\mathcal{V}$, according to (4.9), i.e. (4.11), respectively. The relation of these sets is shown graphically in Figure 4.1 in Chapter 4.

The purpose of the input u is to implement a stabilizing feedback. A helpful property is that $\mathcal{U}_d$, i.e. $\mathcal{U}$, contains an arbitrarily small neighborhood of the origin. Formally, the set $\mathcal{U} \subset \mathcal{U}_d$ is required to contain an arbitrarily small ball $\mathcal{B}$ (4.12)

$$\mathcal{B} \subset \mathcal{U} = \mathcal{V} - \bar{u}, \tag{5.6}$$

Since the DC link voltage is assumed to be positive definite, the set $\mathcal{V}$ is nonempty. Moreover, the ball $\mathcal{B}$ can be chosen arbitrarily small such that $\mathcal{B} \subset \mathcal{V}$ is always true. Consequently, (5.6) translates to a condition on the feedforward controller $\bar{u}$, which is specified by Proposition 4.1. In other words, the magnitude of the feedforward controller must be bounded and satisfy the condition

$$\bar{u} \in \operatorname{int} \mathcal{V}. \tag{5.7}$$

The vector $\bar{u}$ is a function of the reference $\bar{r}_{\alpha\beta}$ and depends on the system parameter ω. Thus, Proposition 4.1 implicitly defines, which flux references $\bar{r}_{\alpha\beta}$, i.e. and $\bar{r}_{dq}$, are allowed and can be achieved by the system. A simple condition is provided by Proposition 4.2, which states that $\bar{u} \in \operatorname{int} \mathcal{V}$ if

$$|\omega| \|\bar{r}_{dq}\| = |\omega| \|\bar{r}_{\alpha\beta}\| \leq \rho_v v_r \tag{5.8}$$

where $\rho_v \in (0, 1)$ is the voltage safety factor used to generate the references. As a consequence, the references, which are obtained with the reference generation procedure of Section 2.4, satisfy these conditions.

5.2. Stabilizability

By Artstein's theorem [3], [91], the existence of a regular stabilizing controller $\kappa(x)$ is strongly linked to control Lyapunov functions. A function $\Gamma(x)$ ($\Gamma(x) > 0$ for all $x \neq 0$; $\Gamma(0) = 0$; $\Gamma(x)$ is radially unbounded and continuous in the origin) is said to be a global control Lyapunov function iff $0 \in \mathcal{U}/\mathcal{U}_d$ and

$$\exists u \in \mathcal{U}/\mathcal{U}_d \ : \ \Delta\Gamma(x, u) = \Gamma(x + T_s u) - \Gamma(x) < 0 \quad \forall x \neq 0. \tag{5.9}$$

In other words, control Lyapunov functions deal with the existence of an admissible stabilizing feedback $u \in \mathcal{U}/\mathcal{U}_d$. By Theorem 4.1, the stability of the nonlinear controller $\kappa(x)$ is proven using Lyapunov functions. These functions can be used as control Lyapunov functions, to achieve and show stability of CCS-MPC.

Proposition 5.1. *If $\bar{u} \in \operatorname{int} \mathcal{V}$, then $\Gamma(x) \overset{def}{=} \|x\|_g$ with $g \in \{1, 2, \infty\}$ is a control Lyapunov function for the system (5.3b), i.e. $0 \in \mathcal{U}/\mathcal{U}_d$ and*

$$\exists u \in \mathcal{U}/\mathcal{U}_d \ : \ \Delta\Gamma(x, u) = \|x + T_s u\|_g - \|x\|_g < 0 \quad \forall x \neq 0. \tag{5.10}$$

Proof. The function $\Gamma(x) \overset{\text{def}}{=} \|x\|_g$ with $g \in \{1, 2, \infty\}$ satisfies $\Gamma(x) > 0$ for all $x \neq 0$, $\Gamma(0) = 0$, and $\Gamma(x)$ is radially unbounded and continuous in the origin. If $\bar{u} \in \operatorname{int} \mathcal{V}$, then $0 \in \mathcal{U}/\mathcal{U}_d$ by Proposition 4.1 and $\kappa(x) \in \mathcal{U}/\mathcal{U}_d$ by Proposition 4.3. Thus, $u = \kappa(x) \in \mathcal{U}/\mathcal{U}_d$ is an admissible control input and leads to $\Delta\Gamma(x, \kappa(x)) < 0$ for all $x \neq 0$ by Theorem 4.1. □

Control Lyapunov functions are useful to show the existence of a stabilizing control input $u \in \mathcal{U}/\mathcal{U}_d$. If the Proposition 5.1 holds, an input $u \in \mathcal{U}/\mathcal{U}_d$ can always be found, which satisfies the following statement.

Corollary 5.1. *An admissible $u \in \mathcal{U}/\mathcal{U}_d$ is stabilizing if it satisfies*

$$\Delta\Gamma(x,u) < 0, \text{ or} \tag{5.11a}$$
$$\Gamma(x+T_s u) < \Gamma(x), \text{ or} \tag{5.11b}$$
$$\|x+T_s u\|_g < \|x\|_g, \tag{5.11c}$$

for all $x \neq 0$ and $u = 0$ otherwise.

A more restrictive[1] way to write this requirement is that $u \in \mathcal{U}/\mathcal{U}_d$ provides at least the stability properties of $\kappa(x_0)$, which yields

Corollary 5.2. *An admissible $u \in \mathcal{U}/\mathcal{U}_d$ is stabilizing and has at least the stability properties of $\kappa(x_0)$ if*

$$\Delta\Gamma(x,u) \leq \Delta\Gamma(x_0,\kappa(x)), \text{ or} \tag{5.12a}$$
$$\Gamma(x+T_s u) \leq c(x), \tag{5.12b}$$

with $\Gamma(x+T_s u) = \|x+T_s u\|_g$ and $c(x) \overset{def}{=} \|x+T_s\kappa(x)\|_g$.

If a control Lyapunov function, i.e. the Corollary 5.1 or Corollary 5.2, is persistently satisfied (satisfied for all time) then the closed loop system is stable.

5.3. Constrained Finite Time Optimal Control (CFTOC)

Model Predictive Control (MPC) is defined by an optimization problem, which is called the *Constrained Finite Time Optimal Control* (CFTOC) problem [19], [61]. The control goals are defined by a cost function $J(.)$ taking $N \in \mathbb{N}_{>0}$ future time steps into account, which are called the *prediction horizon.* In this text, the Lyapunov-based MPC is used, which yields the CFTOC [26], [68]

$$\underset{u_0,\ldots,u_{N-1}}{\text{minimize}} \; J(x_0,u_0,\ldots,u_{N-1}) \tag{5.13a}$$
$$\text{subject to } x_{j+1} = x_j + T_s u_j; \tag{5.13b}$$
$$u_j \in \mathcal{U}_j/\mathcal{U}_{d,j} \overset{\text{def}}{=} \mathcal{V}/\mathcal{V}_d - \bar{u}_j; \tag{5.13c}$$
$$\Gamma(x_0+T_s u_0) \leq c(x_0). \tag{5.13d}$$

[1] The tuning parameter of $\kappa(x)$ is required to satisfy $K \in (0,2)$. However, if it tends to zero, Corollary 5.2 tends to Corollary 5.1. In other words, the quantitative effect of having at least the stability properties of $\kappa(x_0)$ vanishes.

The *cost function* (5.13a) defines the control goals. It is assumed that the cost function $J(.)$ is globally defined on $\mathbb{R}^n \times \mathbb{R}^m \times \cdots \times \mathbb{R}^m$, in particular $|J(.)| < \infty$. This implies that $J(.)$ cannot impose hard constraints, e.g. using barrier functions [59], which need to be replaced with soft constraints, e.g. using penalty functions [59]. This assumption is sufficient that the following theoretical treatment aiming feasibility and stability is sound. However, significantly stronger assumption (e.g. continuity, differentiability, convexity) on $J(.)$ are needed in practice to be able to solve the CFTOC (5.13) efficiently.

The cost function is minimized taking the *constraints* (5.13b), (5.13c), and (5.13d) into account. The constraint (5.13b) introduces the plant dynamic in the optimization problem and (5.13c) requires that the input constraints are satisfied. The notation x_j and u_j is introduced for a prediction regarding sampling time instant $k + j$, $j \in [0, N] \subset \mathbb{N}$, which are carried out at time instant k. It is important to note that the open-loop predictions x_j do *not* correspond with the closed-loop state $x[k + j]$ in general even if the model is exact [19]. The constraint (5.13d) introduces the control Lyapunov constraint. It restricts the choice of the first input u_0 such that u_0 is stabilizing according to Corollary 5.2. This constraint is introduced to achieve stability of MPC without adding further requirements on the cost function nor the constraints.

The CFTOC problem is parametrized with $x_0 \overset{\text{def}}{=} x[k]$ and $\bar{u}_j$ in (5.13c). Requiring that there exists a stabilizing controller for each prediction step by Proposition 4.3, results in the following assumption.

Assumption 5.1. The sequence $\bar{u}_0, \ldots, \bar{u}_{N-1}$ satisfies $\bar{u}_j \in \operatorname{int} \mathcal{V}$ and is independent of the state x_j.

In practice, exact information on how $\bar{u}_j$ evolves is not available in general. A simple approach to estimate $\bar{u}_0, \ldots, \bar{u}_{N-1}$ is to assume that the dq reference r_{dq}, i.e. $\bar{r}_{dq}$, and the velocity ω are approximately constant over the prediction horizon. This approach yields

$$\bar{u}_j \approx -\frac{1}{T_s}(\mathbf{I} - \mathbf{T}_{dq}^{-1}(\omega T_s))\mathbf{T}_{dq}^{-1}(\epsilon + j\omega T_s)\bar{r}_{dq} \tag{5.14a}$$

$$= -\frac{1}{T_s}(\mathbf{I} - \mathbf{T}_{dq}^{-1}(\omega T_s))\mathbf{T}_{dq}^{-1}(j\omega T_s)\bar{r}_{\alpha\beta,0}. \tag{5.14b}$$

Clearly, these approximations use the assumptions $\omega_0 \approx \omega_1 \approx \cdots \approx \omega_N$ and $\bar{r}_{dq,0} \approx \bar{r}_{dq,1} \approx \cdots \approx \bar{r}_{dq,N}$, which are stronger than Assumption 5.1. On the other hand, $\bar{u}_0 \in \operatorname{int} \mathcal{V}$ implies $\bar{u}_j \in \operatorname{int} \mathcal{V}$ in this case.

The CFTOC (5.13) is solved by an input sequence

$$\mathrm{U} \overset{\text{def}}{=} [u_0', \dots, u_j', \dots, u_{N-1}']', \tag{5.15}$$

which produces the state sequence

$$\mathrm{X} \overset{\text{def}}{=} [x_1', \dots, x_j', \dots, x_N']'. \tag{5.16}$$

Using the sequence notation, the cost function (5.13a) is written compactly as

$$J(x_0, \mathrm{U}) \overset{\text{def}}{=} J(x_0, u_0, \dots, u_{N-1}). \tag{5.17}$$

The sequences U and X are said to be *feasible*, if they satisfy the CFTOC constraints (5.13b), (5.13c), and (5.13d). The sequences $\mathrm{U}^\star$ and $\mathrm{X}^\star$ are said to be *optimal*, if they are feasible and yield $J(x_0, \mathrm{U}^\star) \leq J(x_0, \mathrm{U})$ for all feasible U. Thus, the CFTOC (5.13) is said to be feasible if at least one feasible sequence U and X can be found. Moreover, it is said to be *persistently feasible* if it is feasible for all future time steps.

Proposition 5.2. *If $\bar{u}_j \in \operatorname{int} \mathcal{V}$, then the CFTOC (5.13) is feasible for all $x_0 \in \mathbb{R}^2$.*

Proof. Feasibility is given if there exists at least one sequence which satisfies (5.13b), (5.13c), and (5.13d). If $\bar{u} \in \operatorname{int} \mathcal{V}$, then $0 \in \mathcal{U}$ by Proposition 4.1 and $\kappa(x) \in \mathcal{U}$ by Proposition 4.3. Thus, the sequence

$$\mathrm{U} \overset{\text{def}}{=} [\kappa(x)', 0', \dots, 0', \dots, 0']', \tag{5.18}$$

is feasible and produces a feasible sequence X (any sequence X is feasible). □

Clearly, the CFTOC is persistently feasible if $\bar{u}_j \in \operatorname{int} \mathcal{V}$ for all future time steps. This results in a condition on the reference $\bar{r}_{\alpha\beta}$, which can be limited appropriately.

5.4. Receding Horizon Policy (RHC)

Model Predictive Control (or *Receding Horizon Control*) uses the receding horizon policy, which is shown in Algorithm 5.1. Simply speaking, the CFTOC (5.13) is solved at the time instant k taking the time steps k to $k+N$ into account. Then, the first input $u_0^\star$ of the optimal input sequence $\mathrm{U}^\star$ is applied to the plant. At the next sampling time instant $k+1$, the computation is repeated taking the time steps $k+1$ to the “receded” horizon $k+N+1$ into account.

First, the parametrization $x[k]$ and $\bar{u}_0, \ldots, \bar{u}_{N-1}$ of the CFTOC (5.13) is measured, i.e. computed. Then, it is checked for integrity. If some $\bar{u}_j \notin \operatorname{int} \mathcal{V}$, the CFTOC (5.13) cannot be solved and no control action $u[k]$ can be generated in general. Consequently, the system has to be stopped. If $\bar{u}_0, \ldots, \bar{u}_{N-1} \in \operatorname{int} \mathcal{V}$, then the CFTOC can be solved according to Proposition 5.2. Once the optimal sequence $\mathrm{U}^\star$ is obtained, the first of the optimal control sequence is applied to the plant $u[k] \leftarrow u_0^\star$. After setting $u[k]$, the algorithm waits for the next sampling instant, when it performs the next iteration. This policy is summarized in Algorithm 5.1.

The receding horizon policy defines an implicit feedback control law, i.e. a closed loop control system. This system can be analyzed for stability. In other words, the stability properties of MPC are studied using the receding control policy Algorithm 5.1 and solving the CFTOC (5.13) at each time step.

Theorem 5.1. *If $\bar{u}_j \in \operatorname{int} \mathcal{V}$ for all future time steps, then the MPC using the receding horizon policy Algorithm 5.1 and solving the CFTOC (5.13) at each time step is stable in the sense of Lyapunov.*

Proof. If $\bar{u}_j \in \operatorname{int} \mathcal{V}$ for all future time steps, the CFTOC (5.13) is persistently feasible by Proposition 5.2. Solving the CFTOC produces an optimal input sequence where the first input $u_0^\star$ is stabilizing by Corollary 5.2. Moreover, MPC implements the (stabilizing) first optimal input $u[k] = u_0^\star$ at each time step (for all future time) according to the receding control policy Algorithm 5.1. Thus, the overall system is stable. □

Remark 5.1. Stability is imposed by (5.13d). Thus, stability is obtained *without* adding further requirements on the cost function (terminal cost), introducing set

Algorithm 5.1: Receding Horizon Policy

```
while true do
    get $x[k]$ and $\bar{u}_0, \ldots, \bar{u}_{N-1}$          /* get parametrization of the CFTOC */
    if $\bar{u}_0, \ldots, \bar{u}_{N-1} \in \operatorname{int} \mathcal{V}$ then
        /* the CFTOC is feasible                                      */
        solve the CFTOC     /* obtain the optimal sequences $\mathrm{U}^\star$ and $\mathrm{X}^\star$ */
        apply $u[k] \leftarrow u_0^\star$  /* apply the first optimal input to the plant */
    else
        /* Some $\bar{u}_j \notin \operatorname{int} \mathcal{V}$; the CFTOC is infeasible in general        */
        stop
    wait for next sampling instant
```

constraints (terminal set), nor the prediction horizon (generally a sufficiently long prediction horizon is required to achieve a sufficiently large feasible initial set, i.e. region of attraction). Such requirements are typically necessary to achieve the feasibility and stability of MPC [15], [16], [19], [52], [61], [64].

5.5. Cost Function

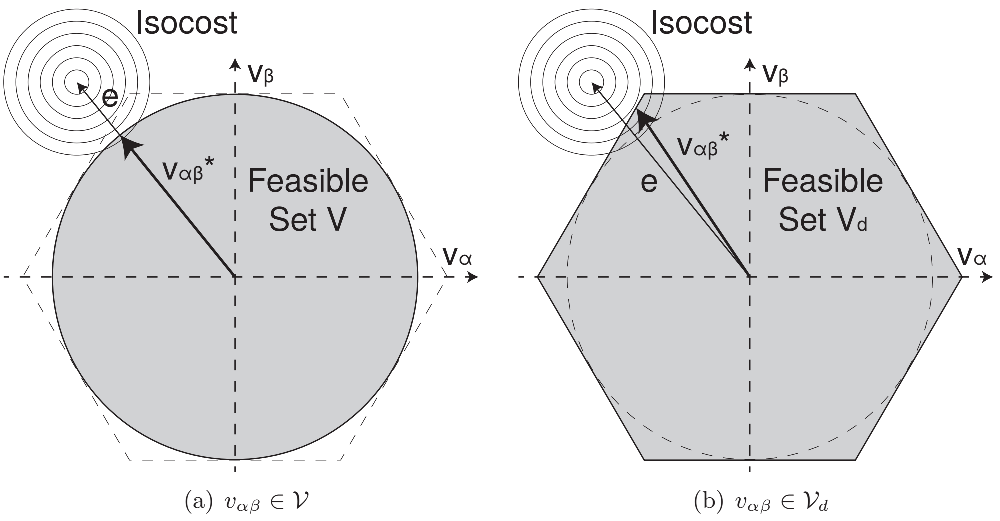
To complete the CCS-MPC design, a cost function needs to be chosen for the CFTOC (5.13). The cost function is the mathematical tool to define the control goal by penalizing undesired behavior. In principle, an arbitrary cost function (defined on $\mathbb{R}^n \times \mathbb{R}^m \times \cdots \times \mathbb{R}^m$) can be chosen without affecting the stability properties provided by Theorem 5.1. In practice, an advantageous cost function, which supports the convergence towards the origin, is chosen. A simple and effective cost function for linear systems is [19]

$$J(x_0, u_0, \dots, u_{N-1}) \overset{\text{def}}{=} \sum_{j=1}^{N} \|\mathbf{Q}x_j\|_q + \gamma \sum_{j=0}^{N-1} \|\mathbf{R}u_j\|_q, \tag{5.19}$$

where $q \in \{1, 2, \infty\}$. The scalar $q = 1$ and $q = \infty$ denotes the one-norm, e.g. $\|x_j\|_1$, and the infinity-norm, e.g. $\|x_j\|_\infty$, respectively. Moreover, $q = 2$ denotes the squared Euclidean norm $\|x_j\|_2 = (\|x_j\|)^2$.

The matrices $\mathbf{Q}$ and $\mathbf{R}$ are the tuning parameters of MPC which define the importance of minimizing the state x with respect to having a large control input u. In drive systems, applying a large input (voltage variation) does not come along with drawbacks, e.g. energy consumption. Thus, the typical control goal is obtaining the highest possible dynamics, i.e. to minimize the control error as fast as possible. Formally, this statement is obtained setting $\mathbf{R} = 0$, which means not penalizing the actuation of u. Moreover, the components of the state, i.e. the α and β flux error, should be weighted equivalently and $\mathbf{Q} = \mathbf{I}$ is chosen. Thus, the CFTOC (5.13) becomes

$$\begin{aligned} \underset{u_0,\dots,u_{N-1}}{\text{minimize}} \; & \sum_{j=1}^{N} \|x_j\|_q && (5.20a) \\ \text{subject to } & x_{j+1} = x_j + T_s u_j; && (5.20b) \\ & u_j \in \mathcal{U}_j / \mathcal{U}_{d,j}; && (5.20c) \\ & \Gamma(x_0 + T_s u_0) \leq c(x_0). && (5.20d) \end{aligned}$$



(a) $v_{\alpha\beta} \in \mathcal{V}$

(b) $v_{\alpha\beta} \in \mathcal{V}_d$

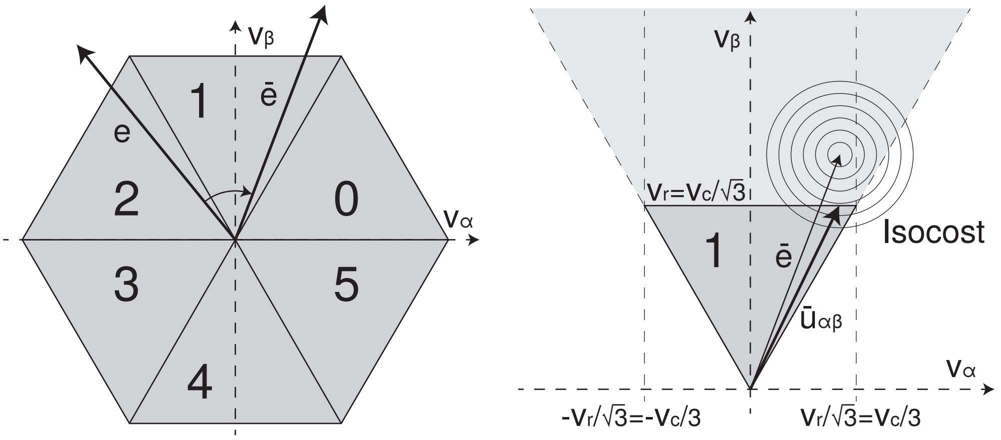
Figure 5.1.: Projection of e onto the feasible set $\mathcal{V}/\mathcal{V}_d$, i.e. solving (5.23)

5.6. Implementation

Several approaches are available to solve the CFTOC (5.20). A real-time implementation needs to be able to solve the CFTOC (5.13), i.e. CFTOC (5.20), within a sampling period such that the input can be applied to the system at the end of the period.

One approach is efficient general purpose numeric solvers, e.g. [35], [45], [67], where the efficient implementations are based on the primal-dual interior-point method [59]. However, numeric solvers are often not available for a specific target platform or can be ported only with difficulties since they rely on libraries e.g. for linear algebra. Specific (primal-dual interior-point) code generation tools, e.g. [30], [63], [99], have been developed to overcome this limitation. However, the primal-dual interior-point method has the inherit limitation that it is difficult to warm-start. Moreover, the optimal solution (or a reasonably close one) must be found such that the proposed solution is feasible in general. This disqualifies the method in most power electronic applications since the available computation capabilities of embedded hardware do not allow to find the optimal solution within a sampling period.

Another numeric solver is the primal fast-gradient method [83], [85], [86]. This method has a lower convergence rate than the primal-dual interior-point method (linear instead of quadratic). Thus, it is less suitable for general purpose medium and large scale optimization problems. Also this method cannot provide a result of



(a) Rotation to sector 1

(b) Projection

Figure 5.2.: Projection of e onto the symmetric polytope $\mathcal{V}_d$

an optimization problem on embedded hardware within power electronic sampling times in general. However, it has some advantages over other numeric solvers, which has attracted the attention of control research. It is simple to combine the *warm start* and *early termination* technique [83]. In other words, the solver can start form an arbitrary initial guess and issues a result of the optimization problem after a given amount of time, i.e. number of solver iterations. The provided result can be arbitrarily bad in terms of optimality but is guaranteed to be feasible. Moreover, the gradient and fast-gradient method can be written as *descent* method. Thus, the result provided by each solver iteration is at least as good as the previous result in terms of optimality.

The third approach, which is introduced, solves the problem parametrically, e.g. using explicit MPC [55], [79]. The idea behind this approach is to compute and store a parametrized solution offline. Thus, the online computation effort is reduced to find the correct solution in memory. This approach is suitable to solve CFTOC (5.13), i.e. CFTOC (5.20), if the cost function is linear or quadratic and the constraints are affine. However, the parameter space of the problem expands fast with the prediction horizon due to the required feedforward controller, which varies over time. As a consequence, the offline and online computation demand to find the solution and the memory requirement to store the controller do not scale well with N. Thus, the method is suitable for implementations with a small prediction horizon N, which are typically a few time steps.

Although general approaches are available, the implementation of MPC with long prediction horizon is typically difficult due to computation constraints [57]. Thus, MPC for power electronic and drive systems is often implemented with a minimum problem size with the prediction horizon $N = 1$, which is sometimes called one-step-ahead prediction [87]. In this section, a reference implementation for this reduced problem is proposed. The compact CFTOC is obtained using the cost function

$$J(x_0, u_0) = \|x_1\|_q. \tag{5.21}$$

A benefit of this formulation is that the control Lyapunov constraint (5.20d) is automatically satisfied and becomes redundant. In other words, $\Gamma(x_1^\star) = \|x_1^\star\|_q = \|x_0 + T_s u_0^\star\|_q \leq c(x_0)$ is true since the optimal cost $J(x_0, u_0^\star) = \|x_1^\star\|_q$ is minimum. Therefore, $u_0^\star$ is stabilizing. Consequently, the one-step-ahead-prediction CFTOC is written as

$$\underset{u \in \mathcal{U}/\mathcal{U}_d}{\text{minimize}} \ \|x + T_s u\|_q. \tag{5.22}$$

This problem can be solved by a simple and efficient algorithm. Using a quadratic cost function $q = 2$, the solution can be computed directly without numerical solver. First, the CFTOC (5.22) is brought into the $v_{\alpha\beta}$ space

$$\underset{v_{\alpha\beta} \in \mathcal{V}/\mathcal{V}_d}{\text{minimize}} \ \|v_{\alpha\beta} - e\|_2, \tag{5.23}$$

with the parametrization $e = \bar{u} - \frac{1}{T_s}x$. The resulting problem (5.23) is known as projection operation, i.e. projecting the vector e onto the feasible set $\mathcal{V}/\mathcal{V}_d$ [20]. This problem is shown in Figure 5.1, where the isocost levels are circles centered in e.

Algorithm 5.2: Algorithm to solve (5.22) for $u \in \mathcal{U}$

```
if $\|\bar{u}\| < v_r$ then
    /* $\bar{u} \in \operatorname{int}\mathcal{V}$; (5.22) has a solution                    */
    $v_{\alpha\beta} \leftarrow e$                                  /* initialize projection */
    /* project onto feasible set                                      */
    if $\|v_{\alpha\beta}\| > v_r$ then
        $v_{\alpha\beta} \leftarrow \frac{v_r}{\|v_{\alpha\beta}\|} v_{\alpha\beta}$
    $v_{\alpha\beta}^\star \leftarrow v_{\alpha\beta}$                                 /* assign optimal result */
else
    /* $\bar{u} \notin \operatorname{int}\mathcal{V}$; (5.22) may not have a solution          */
    stop
```

The problem (5.23) is simple to solve for the quadratic constraint $v_{\alpha\beta} \in \mathcal{V}$ since the constraint is another circle centered in the origin, see Figure 5.1(a). In this case, the optimal $v^\star_{\alpha\beta}$ is given by truncating the vector e with

$$v^\star_{\alpha\beta} = \begin{cases} e & \text{if } \|e\| \le v_r \\ \frac{v_r}{\|e\|} e & \text{otherwise.} \end{cases} \tag{5.24}$$

The policy to compute (5.22) with the constrained $u \in \mathcal{U}$ is summarized in Algorithm 5.2.

Clearly, the error can be further reduced using the larger constraint $v_{\alpha\beta} \in \mathcal{V}_d$, but projecting onto a polytope is more demanding in general [20]. However, the problem can be solved efficiently using the small dimension and symmetries of $\mathcal{V}_d$, which is portrayed in Figure 5.2. The vector e is rotated onto the sector 1 and the new vector $\bar{e}$ is defined with $\|\bar{e}\| = \|e\|$ and $\angle\bar{e} = \angle e - (sec + 1)\ \pi/3$ as is shown in

Algorithm 5.3: Algorithm to solve (5.22) for $u \in \mathcal{U}_d$

```
if $\|\bar{u}\| < v_r$ then
    /* $\bar{u} \in \operatorname{int}\mathcal{V}$; (5.22) has a solution                    */
    $ang \leftarrow \operatorname{atan2} e$                                   /* angle of $e$ */
    $sec \leftarrow \operatorname{floor} \frac{ang}{\pi/3}$                                  /* sector of $e$ */
    $\bar{v}_{\alpha\beta} \leftarrow \bar{e} \leftarrow \begin{bmatrix} \cos\frac{\pi}{3}(sec-1) & \sin\frac{\pi}{3}(sec-1) \\ -\sin\frac{\pi}{3}(sec-1) & \cos\frac{\pi}{3}(sec-1) \end{bmatrix} e$      /* init. projection */
    /* project onto feasible set                                            */
    if $\bar{v}_\alpha > v_r$ then
        $\bar{v}_\alpha \leftarrow v_r$
    else
        if $\bar{v}_\beta > \frac{v_r}{\sqrt{3}}$ then
            $\bar{v}_\beta \leftarrow \frac{v_r}{\sqrt{3}}$
        if $\bar{v}_\beta < -\frac{v_r}{\sqrt{3}}$ then
            $\bar{v}_\beta \leftarrow -\frac{v_r}{\sqrt{3}}$
    $v^\star_{\alpha\beta} \leftarrow \begin{bmatrix} \cos\frac{\pi}{3}(sec-1) & -\sin\frac{\pi}{3}(sec-1) \\ \sin\frac{\pi}{3}(sec-1) & \cos\frac{\pi}{3}(sec-1) \end{bmatrix} \bar{v}_{\alpha\beta}$  /* assign optimal result */
else
    /* $\bar{u} \notin \operatorname{int}\mathcal{V}$; (5.22) may not have a solution              */
    stop
```

Figure 5.2(a). Thus, $\bar{e}$ belongs to the light-gray area in Figure 5.2(b). The projection of $\bar{e}$ onto the feasible sector 1, i.e. the dark-gray area in Figure 5.2(b), is obtained with

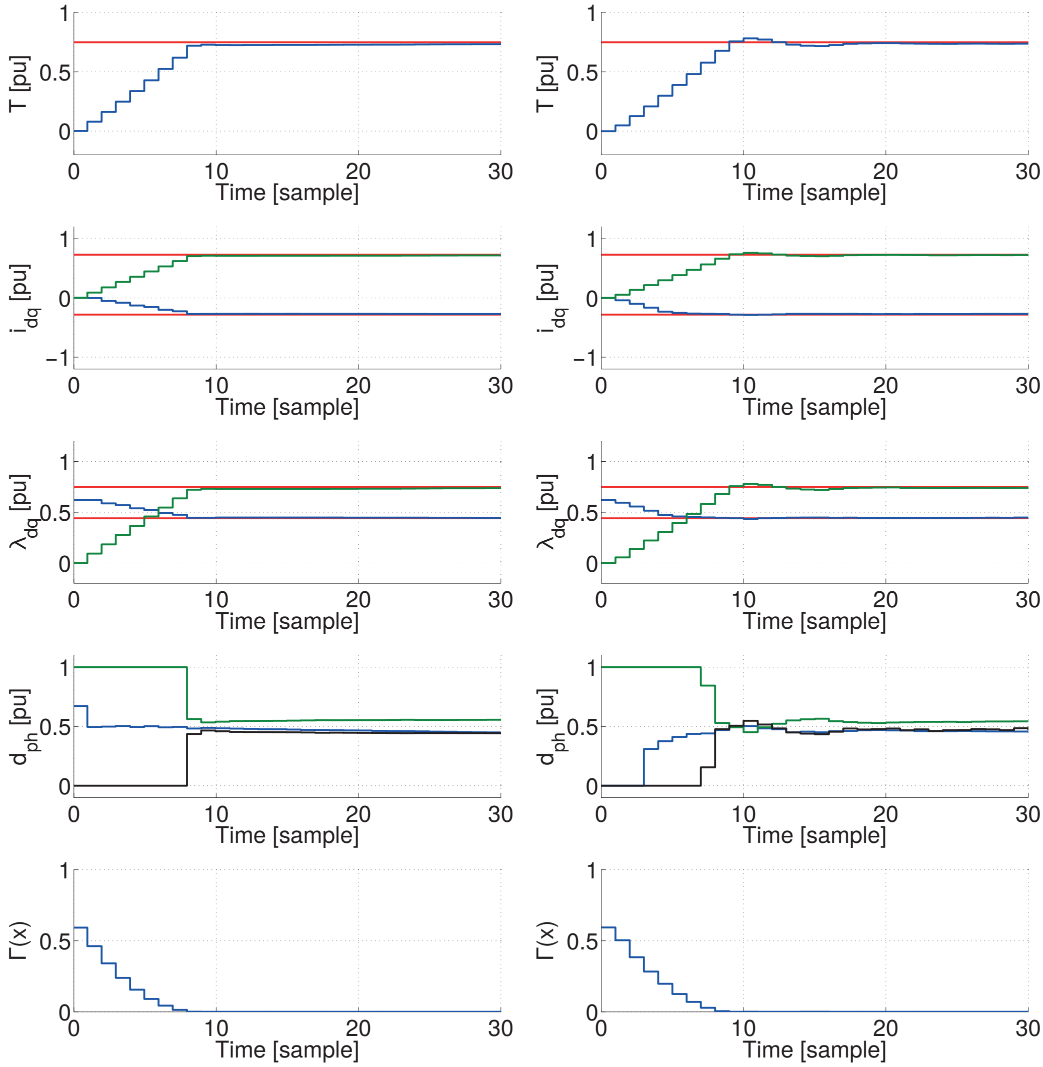
$$\bar{v}^{\star}_{\alpha\beta} = \begin{cases} \bar{e} & \text{if } \bar{e}_\beta \leq v_r; \\ \left[\frac{v_r}{\sqrt{3}}, v_r\right]' & \text{if } \bar{e}_\beta > v_r \text{ and } \bar{e}_\alpha > \frac{v_r}{\sqrt{3}}; \\ \left[-\frac{v_r}{\sqrt{3}}, v_r\right]' & \text{if } \bar{e}_\beta > v_r \text{ and } \bar{e}_\alpha < -\frac{v_r}{\sqrt{3}}; \\ [\bar{e}_\alpha, v_r]' & \text{otherwise.} \end{cases} \tag{5.25}$$

The final $v^{\star}_{\alpha\beta}$ is obtained rotating $\bar{v}^{\star}_{\alpha\beta}$ backward onto the original sector, i.e. $\|v^{\star}_{\alpha\beta}\| = \|\bar{v}^{\star}_{\alpha\beta}\|$ and $\angle v^{\star}_{\alpha\beta} = \angle \bar{v}^{\star}_{\alpha\beta} + (sec+1)\ \pi/3$. The policy to compute (5.22) with the constrained $u \in \mathcal{U}_d$ is summarized in Algorithm 5.3.

To gain sensitivity on the computation requirements of the proposed algorithms, the execution times are compared on the simulation and experimentation platform outlined in Appendix D. The nonlinear controller, i.e. Algorithm 4.1 and Algorithm 4.2, is evaluated as well as CCS-MPC with prediction horizon $N = 1$, i.e. Algorithm 5.2 and Algorithm 5.3. The results are shown in Table 5.1. The nonlinear controller is known to be implementable on experimental hardware since its computation requirements are comparable with e.g. anti-windup proportional-integral control. Since Algorithm 5.2 has lower computation requirements than the nonlinear one, this implementation is considered to be efficient. The execution time of Algorithm 5.3 increases due to the execution of several trigonometric operations but is still acceptable in most cases.

Table 5.1.: Execution times of nonlinear control and CCS-MPC with $N = 1$

Controller	Constraint	Platform	MIN	MEAN	MAX
Nonlinear	$u \in \mathcal{U}$	pc	$4.07\mu s$	$4.34\mu s$	$5.84\mu s$
Nonlinear	$u \in \mathcal{U}_d$	pc	$5.73\mu s$	$5.96\mu s$	$6.85\mu s$
CCS-MPC	$u \in \mathcal{U}$	pc	$3.48\mu s$	$3.69\mu s$	$4.67\mu s$
CCS-MPC	$u \in \mathcal{U}_d$	pc	$6.38\mu s$	$6.72\mu s$	$8.36\mu s$
Nonlinear	$u \in \mathcal{U}$	embedded	$2.53\mu s$	$2.65\mu s$	$2.97\mu s$
Nonlinear	$u \in \mathcal{U}_d$	embedded	$2.53\mu s$	$2.65\mu s$	$3.05\mu s$
CCS-MPC	$u \in \mathcal{U}$	embedded	$1.16\mu s$	$1.20\mu s$	$1.56\mu s$
CCS-MPC	$u \in \mathcal{U}_d$	embedded	$9.90\mu s$	$13.34\mu s$	$15.64\mu s$



(a) Simulation

(b) Experimentation

Figure 5.3.: Torque step from $0pu$ to $0.75pu$ using CCS-MPC

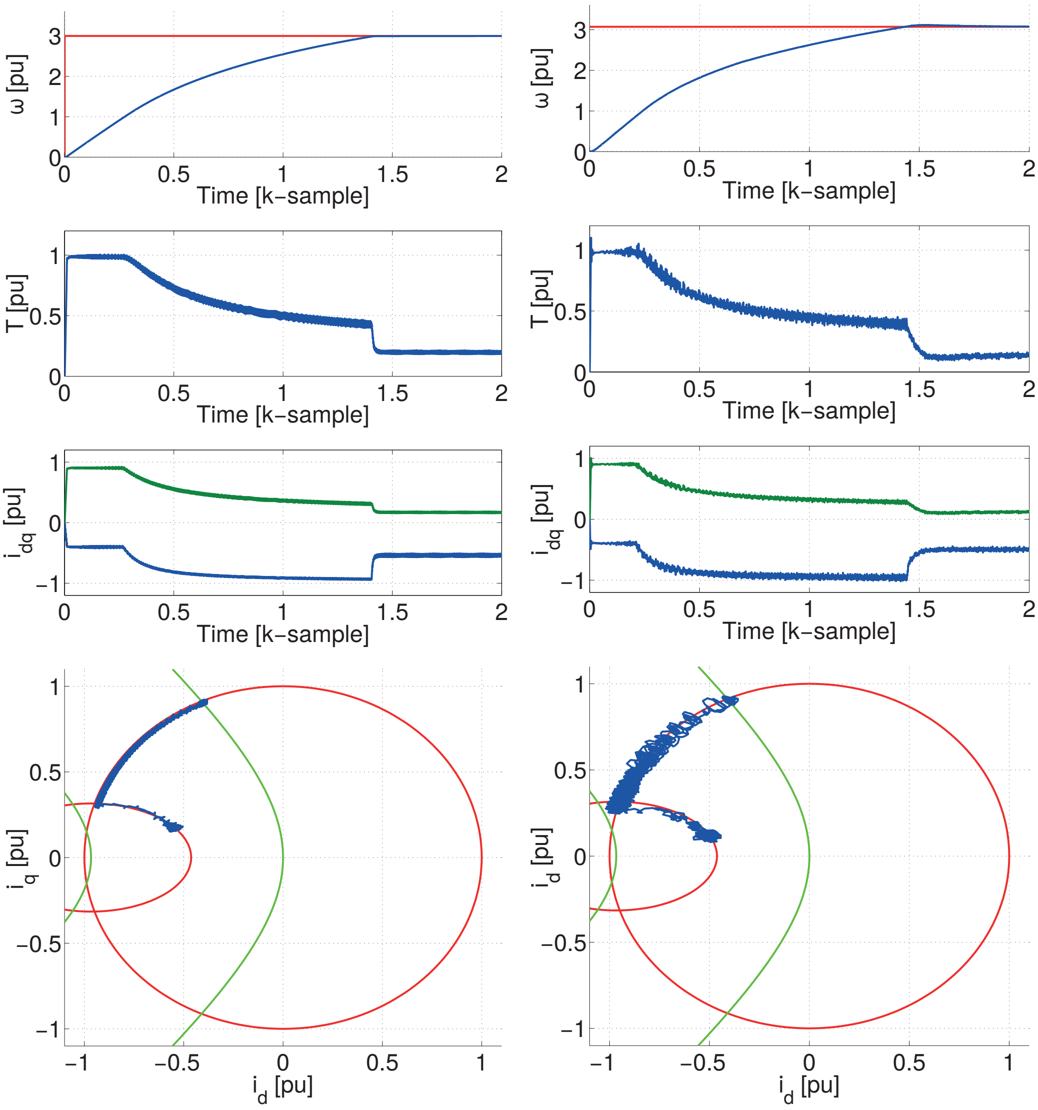
5.7. Dynamic Operation

In this section, the proposed CCS-MPC with prediction horizon $N = 1$ and optimal reference generation procedure is reviewed in dynamic operation. The case $N = 1$ is chosen since it provides the minimum implementation. Moreover, a controller with longer horizon tends to perform better since optimality with respect to a longer horizon is achieved and a controller tends to infinite-time MPC extending the horizon [19]. First, a torque reference step is applied to the reference generation procedure. This results in both a d and q current, i.e. flux, reference step. The step is applied at standstill $\omega = 0$ and during the period of observation, the machine speed remains approximately zero. The simulation and experimental results are shown in Figure 5.3. From top, the figure shows the reference (R, red) and actual (T, blue) torque; the reference current vector (r_{dq}, red), the actual d (i_d, blue) and q (i_q, green) current; the reference flux vector ($\bar{r}_{dq}$, red), the actual d (λ_d, blue) and q (λ_q, green) flux; the duty cycle d_{ph}; and the Lyapunov function $\Gamma(x)$ with $x = \lambda_{\alpha\beta} - \bar{r}_{\alpha\beta}$.

The results confirm the CCS-MPC design and implementation. They show that the Lyapunov function is strictly decreasing once the reference step is applied. The controller applies the maximum voltage according to the input constraint $\mathcal{U}_d$ until the reference is reached. As a consequence the raise time depends on the electric angle, which defines the voltage magnitude in a certain direction. However, the raise time is at least as good as using the input constraint $\mathcal{U}$, where the voltage magnitude is constant. Comparing simulation and experimentation, an overshoot is observed on the latter results. This effect is introduced by the observer, which is treated in Appendix C. The observer converges slower than the controller and relies on integration in presence of model uncertainties.

Moreover, a speed reference step from standstill to $3pu$ is evaluated using an anti-windup proportional-integral speed controller. Since the reference step is large, the speed loop saturates and the maximum available torque is applied approximately until the speed reference is reached. The results are presented in Figure 5.4. From top, the figure shows the reference (red) and actual (ω, blue) speed; the torque (T); the actual d (i_d, blue) and q (i_q, green) current; and the current i_{dq} in the current state-plane.

The results confirm the design and implementation of the CCS-MPC and the reference generation procedure. At low speed, the maximum torque and acceleration are constant. The current producing maximum torque lies on the intersection of the MTPA and the isocurrent locus. Increasing the speed beyond rated speed, this operation point is not available anymore due to the voltage limit. Thus, the current moves along the isocurrent locus until the reference speed is achieved. Then the current moves along the isoflux locus to the operation point which provides the torque to maintain the machine at reference speed.



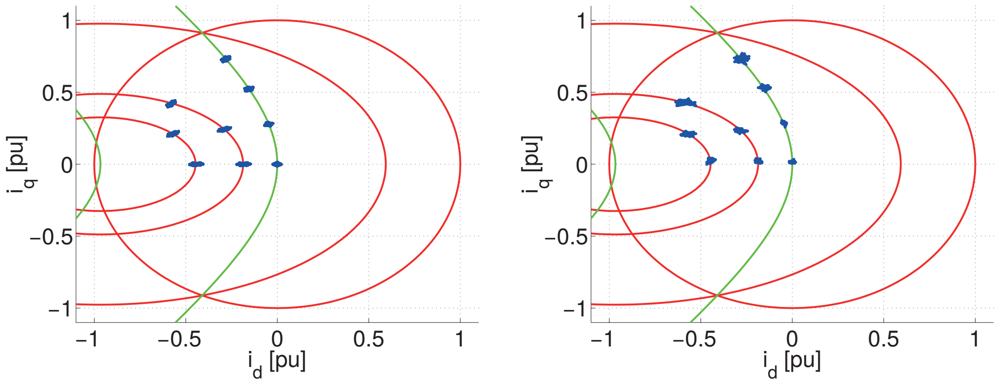
(a) Simulation

(b) Experimentation

Figure 5.4.: Speed step form standstill to $3pu$ using CCS-MPC

5.8. Steady-State Operation

In this section, the proposed CCS-MPC with prediction horizon $N = 1$ and optimal reference generation procedure is reviewed in steady-state operation. First, different



(a) Simulation

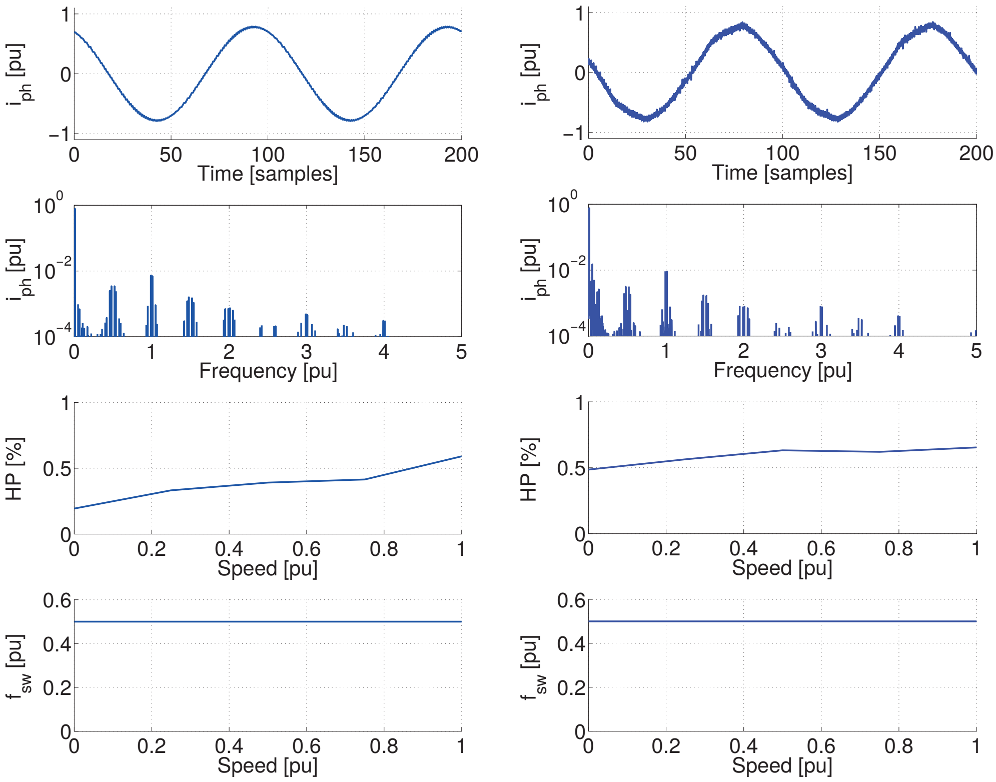
(b) Experimentation

Figure 5.5.: Steady-state operation points using CCS-MPC with horizon $N = 1$ and reference generation procedure

load torques are applied such that the machine produces a constant torque T at different speeds ω. The following cases are evaluated: $\omega = 1.0pu$ with $T = \{0pu, 0.25pu, 0.5pu, 0.75pu\}$; $\omega = 2.0pu$ with $T = \{0pu, 0.25pu, 0.5pu\}$; and $\omega = 3.0pu$ with $T = \{0pu, 0.25pu\}$. The simulation and experimental results are shown in Figure 5.5. Ideally, steady-state operation with given conditions (torque, speed) would lead to a single point on the state plane. In practice, the states stay in the neighborhood of this point due to power converter switching, measurement noise, etc. The results show that the drive system applies states according to the reference generation procedure. The controller operates on the MTPA trajectory at $\omega = 1.0pu$. At $\omega = 2.0pu$ and $\omega = 3.0pu$, the system operates on the isoflux locus, which is defined by the speed ω.

Finally, the steady-state current quality, which is achieved by the CCS-MPC with horizon $N = 1$ is evaluated. Using a modulator, the switching related harmonics depend mainly on the underlying modulation scheme. Figure 5.6 and Figure 5.7 depict the results using the symmetric SVM (SSVM) and the discontinuous SVM (DSVM), respectively. From top, the typical current waveform and spectrum is presented (at $0.75pu$ speed and $0.75pu$ torque). In these figures, frequencies are normalized with the sampling frequency $f_s = 1/T_s$. Moreover, the harmonic power (HP) and the average switching frequency f_{sw} is evaluated as a function of the machine speed.

The current harmonics of a drive system have multiple origins. Low frequency harmonics of a drive system are introduced by the electrical machine, e.g. due to harmonics in the back-EMF. In this section, the focus lies on harmonics, which are



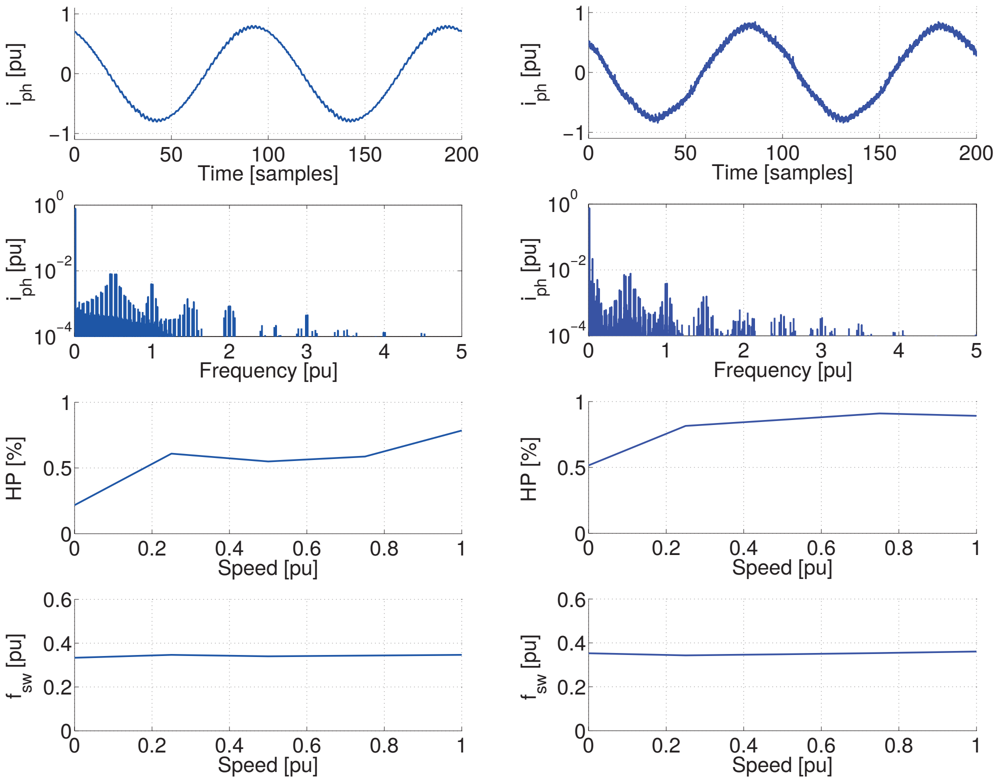
(a) Simulation

(b) Experimentation

Figure 5.6.: Current quality using CCS-MPC with horizon $N = 1$ and SSVM

introduced by control and switching. Thus, the performance index *Harmonic Power* (HP), which is defined by (4.27), is used. It captures the harmonics from interest but disregards low frequency machine specific harmonics evaluating the frequency band from $0.1f_s$ to $10f_s$. The index is proportional to the power of the harmonics in the defined harmonic band as average of the three phases. The switching frequency is evaluated counting the number of switch transitions per phase $n_{ph} \in \mathbb{N}^3$ in an evaluation interval $T_{sw} \in \mathbb{R}_+$ and the average switching frequency is computed according to (4.29).

Both SSVM and DSVM exhibit significant switching harmonics in the neighborhood of the PWM carrier frequency [65] and its multiples. The carrier frequency is half the sampling frequency since the drive system is sampled at the positive and negative peak of the triangular carrier frequency. In SSVM, the carrier frequency corresponds to the average switching frequency leading to distinct switching harmonics and



(a) Simulation

(b) Experimentation

Figure 5.7.: Current quality using CCS-MPC with horizon $N = 1$ and DSVM

low noise. In DSVM, the switching frequency is reduced to approximately 2/3 of the carrier frequency decreasing the current quality, i.e. increasing the HP. Both modulation concepts achieve a constant switching frequency, which is independent of the machine speed[2]. Comparing the simulation and experimental results, the current quality deteriorates. The main causes are measurement noise and the presence of machine dependent low order harmonics.

The switching properties of CCS-MPC make the concept interesting for “standard” applications where typically scalar and vector control [42], [47], [49] is used. The current quality and switching frequency can be adopted to the needs varying the carrier frequency and choosing a modulation scheme. In acoustic noise sensitive

[2] Using DSVM, there are sampling instants where three legs are switched instead of two. The number of such instants is proportional to the fundamental frequency of the current such that the switching frequency is only approximately independent from the machine speed.

applications, CCS-MPC may need to be combined with random PWM strategies [7], [8] to avoid tones in the acoustic spectrum.

Chapter 6.

Finite Control Set (FCS) MPC

In this chapter, Finite Control Set (FCS) Model Predictive Control (MPC) is designed in the virtual flux space. FCS specifies the input constrained set to be an integer set, which is obtained using direct actuation. Stability is achieved using the Lyapunov-based MPC approach, where neither requirement on the prediction horizon nor a terminal set is needed. Efficient algorithms are presented such that the FCS-MPC can be implemented on embedded control hardware meeting typical real time requirements of power electronic systems. The FCS-MPC design is evaluated in simulation and on an experimental test bench.

6.1. System

In this section, the system dynamics and constraints are defined. Since the system dynamics are the same than the ones defined in Chapter 4 and Chapter 5, only a summary is provided here. The model is based on the discrete-time state-space formulation the in $\alpha\beta$ reference frame (1.19)

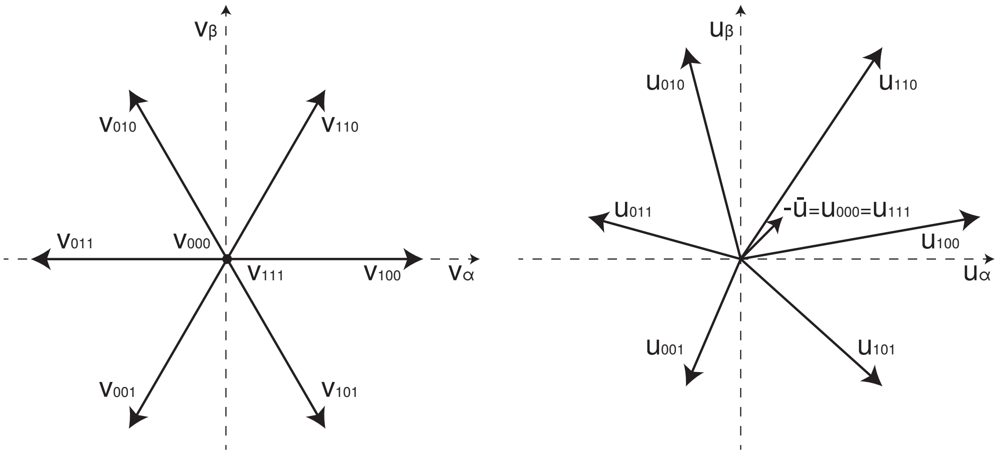
$$\lambda_{\alpha\beta}[k+1] = \lambda_{\alpha\beta}[k] + T_s v_{\alpha\beta}[k], \tag{6.1}$$

assuming that disturbances (resistive voltage drop, inverter nonlinear effects, etc.) are compensated externally. The control goal is to track an $\alpha\beta$ reference vector with the dynamics

$$\bar{r}_{\alpha\beta}[k+1] = \mathbf{T}_{dq}^{-1}(\omega[k]T_s)\bar{r}_{\alpha\beta}[k]. \tag{6.2}$$

The reference tracking problem is transformed into a regulation problem using the state $x = \lambda_{\alpha\beta} - \bar{r}_{\alpha\beta}$, which yields

$$x[k+1] = x[k] + T_s u[k], \tag{6.3}$$



(a) Set $\mathcal{V}_s$

(b) Set $\mathcal{U}_s$

Figure 6.1.: Relation of the sets $\mathcal{V}_s$ and $\mathcal{U}_s$

with the input

$$u[k] = v_{\alpha\beta}[k] - \bar{u}[k], \tag{6.4a}$$

$$\bar{u}[k] = -\frac{1}{T_s}(\mathbf{I} - \mathbf{T}_{dq}^{-1}(\omega[k]T_s))\bar{r}_{\alpha\beta}[k]. \tag{6.4b}$$

The original control input $v_{\alpha\beta}$ is the sum of a *feedforward* controller $\bar{u}$ and a *feedback* controller u.

In this chapter, the inputs $v_{\alpha\beta}$, which are achieved via direct actuation, are considered. These $v_{\alpha\beta}$ lie in the *finite control set* $\mathcal{V}_s$ (3.6), which is a nonconvex integer set. The set $\mathcal{V}_s$ is related to the convex control set $\mathcal{V}_d$ (3.9) and $\mathcal{V}$ (3.10). The set $\mathcal{V}_d$ is the convex hull [19], [20] of $\mathcal{V}_s$, i.e. $\mathcal{V}_d = \operatorname{hull} \mathcal{V}_s$ and $\mathcal{V}$ is the Chebyshev ball [19], [20] of $\mathcal{V}_d$, i.e. $\mathcal{V} = \operatorname{ball} \mathcal{V}_d$. The requirements on the input can be written as the following equivalent statements

$$v_{\alpha\beta} \in \mathcal{V}_s \quad \Leftrightarrow \quad u + \bar{u} \in \mathcal{V}_s \quad \Leftrightarrow \quad u \in \mathcal{U}_s. \tag{6.5}$$

The set $\mathcal{V}_s$ is defined by (3.6)

$$\mathcal{V}_s = v_c \mathbf{T}_{\alpha\beta} \circ \{0,1\}^3 = \{v_{000}, v_{100}, \ldots, v_{111}\}. \tag{6.6}$$

The set $\mathcal{U}_s \overset{\text{def}}{=} \mathcal{V}_s - \bar{u}$ is the $-\bar{u}$ displaced set $\mathcal{V}_s$. The relation of $\mathcal{V}_s$ and $\mathcal{U}_s$ is shown graphically in Figure 6.1.

The purpose of the input u is to implement a stabilizing feedback. Similarly to the continuous control set case, the set $\mathcal{U} = \operatorname{ball\,hull} \mathcal{U}_s$ is required to contain an arbitrarily small ball centered in the origin

$$\mathcal{B} \overset{\text{def}}{=} \left\{ u \in \mathbb{R}^2 \mid \|u\| \leq b \right\} \subset \mathcal{U} = \mathcal{V} - \bar{u}, \tag{6.7}$$

where $b \in \mathbb{R}_{>0}$ is an arbitrarily small positive integer.

Clearly, (6.7) is equivalent to (4.12) and Proposition 4.1 and Proposition 4.2 remain valid. Thus, $\bar{u}$ is required to belong to the interior of $\mathcal{V}$, i.e. $\bar{u} \in \operatorname{int} \mathcal{V}$. In other words, a simple condition is obtained that defines, which references $\bar{r}_{\alpha\beta}$ can be applied to the system as a function of the electrical velocity ω and the DC link voltage v_c, i.e. the rated voltage v_r. By Remark 4.1, a convenient way to choose the references is the reference generation procedure outlined in Section 2.4. Similar to the CCS case, it is possible to relax the requirement (6.7) to $\mathcal{B} \subset \mathcal{U}_d = \mathcal{V}_d - \bar{u}$, i.e. $\bar{u} \in \operatorname{int} \mathcal{V}_d$.

6.2. Stabilizability

In this section, the stabilizability of the system is analyzed. It is observed that the closed loop system *cannot* be stable in the sense of Lyapunov due to the integer set $\mathcal{U}_s$. If a state x is sufficiently close to the origin, there does not exist an input $u \in \mathcal{U}_s$ to further reduce a Lyapunov function in general. The formal incompatibility with Lyapunov theory is that there exists no Lyapunov candidate function, which is continuous in the origin. Thus, the closed loop system is set stable at best. This means the state converges to a set $\mathcal{D}$, which contains the origin $0 \in \mathcal{D}$, and remains ultimately bounded by this set. In this section, the set stabilizability of the system is analyzed. This requires to prove the existence of an input $u \in \mathcal{U}_s$ such that the state converges to $\mathcal{D}$ and remains in that set for all future time.

In fairness, it is observed that also the CCS system is only set stable considering the switching by the modulation scheme. There, the sampled (measured) state is a time average over a sampling period, which is obtained as combination of voltage vectors. Thus, the instantaneous state is subject to a ripple and stays only in a neighborhood of the origin but *not* at the origin. Clearly, the switching is transparent to the control system and is handled by SVM/PWM, which simplifies the control design and implementation. The FCS system does omit the modulator by design. Thus, the controller needs to handle the instantaneous state, i.e. the switching and ripple, and that the system is not able to converge to the origin.

The set stabilizability of the FCS system cannot be shown similar to the CCS system. A simple stabilizing feedback $u = \kappa_s(x) \in \mathcal{U}_s$ does not exist due to the integer nature of $\mathcal{U}_s$. Thus, a different approach is used.

6.2.1. Control Lyapunov Function

For the set stabilizability analysis of the FCS system, a candidate control Lyapunov function $\Gamma(x)$ is introduced. The function has hexagonal sublevel sets Ω, which are similar to (4.9)

$$\Omega = \left\{ u \in \mathbb{R}^2 \mid \mathbf{H}x \leq X_l \right\}, \tag{6.8}$$

where the levels are defined by X_l and matrix $\mathbf{H}$ is

$$\mathbf{H} = \begin{bmatrix} 0 & 1 \\ \sqrt{3}/2 & 1/2 \\ \sqrt{3}/2 & -1/2 \\ 0 & -1 \\ -\sqrt{3}/2 & -1/2 \\ -\sqrt{3}/2 & 1/2 \end{bmatrix}. \tag{6.9}$$

Thus, the candidate control Lyapunov function is defined as

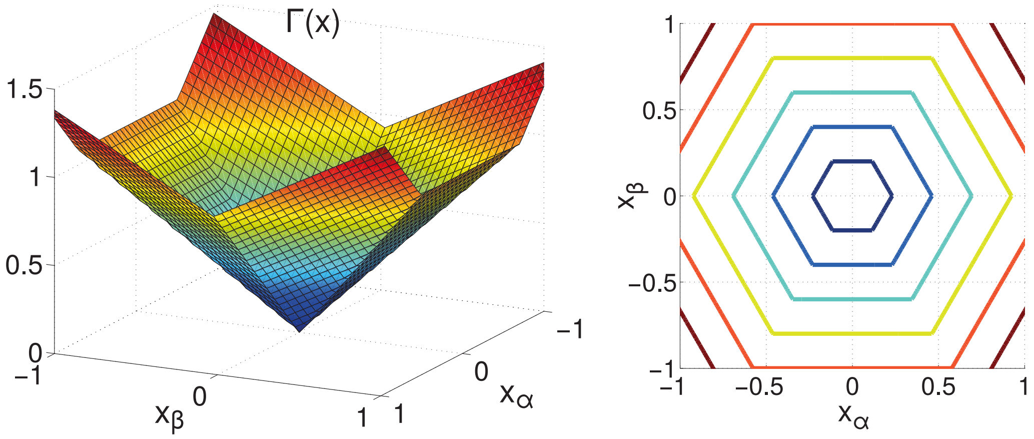
$$\begin{aligned} \Gamma(x) \overset{\text{def}}{=} \quad & \min\nolimits_{X_l} \quad X_l, \\ & \text{subject to} \quad \mathbf{H}x \leq X_l. \end{aligned} \tag{6.10}$$

The Lyapunov function[1] and its sublevel sets are shown in Figure 6.2. According to (6.10), X_l is chosen minimum such that x lies on the border of the polytope $\mathbf{H}x \leq X_l$. The resulting Lyapunov function value can be computed efficiently using the small dimension and the symmetries of the sublevel sets. The vector x is rotated onto the first sector, which yields $\bar{x}$ (similar to the rotation in Figure 5.2(a) in Chapter 5). Then the β component of $\bar{x}$ corresponds to the Lyapunov function value $\Gamma(x) = [0, 1]\bar{x}$. The policy to compute the Lyapunov function (6.10) efficiently is summarized in Algorithm 6.1.

[1] The candidate control Lyapunov function (6.10) is a prism and it has similar properties than e.g. $\|x\|_\infty$, which can be written as

$$\begin{aligned} \|x\|_\infty = \quad & \min\nolimits_{X_l} \quad X_l, \\ & \text{subject to} \quad [\mathbf{I}, -\mathbf{I}]'x \leq X_l, \end{aligned} \tag{6.11}$$

where the function levels are squares instead of regular hexagons.



(a) Function value $\Gamma(x)$

(b) Sublevelsets Ω

Figure 6.2.: Candidate control Lyapunov function $\Gamma(x)$

Using a candidate control Lyapunov function, the set stabilizability of the FCS system can be shown. The state x of the system converges towards the origin as long as the following statement holds

$$\exists u \in \mathcal{U}_s \; : \; \Delta\Gamma(x,u) = \Gamma(x + T_s u) - \Gamma(x) < 0. \tag{6.12}$$

6.2.2. Preset

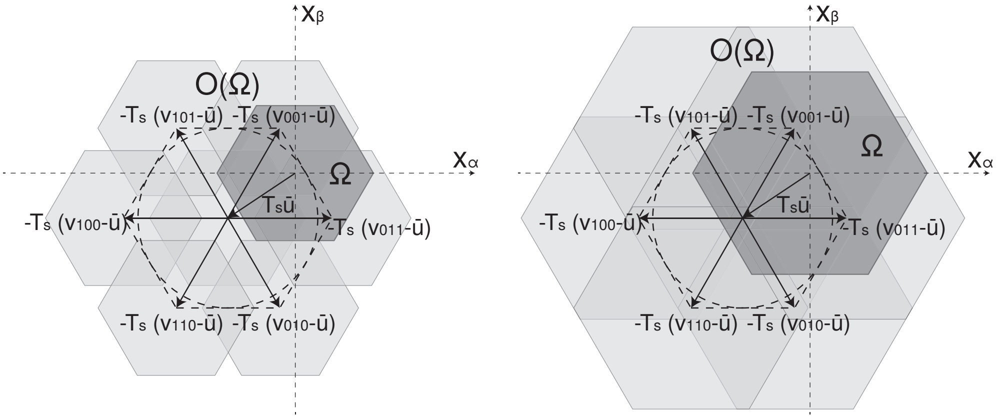
To derive system properties, the concept of *preset* is introduced [19], [52].

Definition 6.1. The *preset* $\mathcal{O}(\Omega)$ is the set of states $x[k] \in \mathbb{R}^2$ which can be driven by an admissible control input $u[k] \in \mathcal{U}_s$ to the set $\Omega \subset \mathbb{R}^2$, i.e. $x[k+1] \in \Omega$. The preset is formally defined as

$$\mathcal{O}(\Omega) \overset{\text{def}}{=} \left\{ x \in \mathbb{R}^2 \mid \exists u \in \mathcal{U}_s \; : \; x + T_s u \in \Omega \right\}. \tag{6.13}$$

Algorithm 6.1: Algorithm to solve (6.10)

```
$ang \leftarrow \operatorname{atan2} x$                                   /* angle of $x$ */
$sec \leftarrow \operatorname{floor} \frac{ang}{\pi/3}$                  /* sector of $x$ */
$\bar{x} \leftarrow \begin{bmatrix} \cos\frac{\pi}{3}(sec-1) & \sin\frac{\pi}{3}(sec-1) \\ -\sin\frac{\pi}{3}(sec-1) & \cos\frac{\pi}{3}(sec-1) \end{bmatrix} x$    /* rotate onto sector 1 */
$\Gamma(x) \leftarrow [0,1]\bar{x}$                                    /* get Lyapunov function value */
```



(a) Ω with $X_l < T_s v_r$

(b) Ω with $X_l \geq T_s v_r$

Figure 6.3.: The preset $\mathcal{O}(\Omega)$

In other words, the preset $\mathcal{O}(\Omega)$ defines the states $x \in \mathbb{R}^2$, which can be steered to the set Ω within one step. Now, the preset is computed for the sublevel set Ω and the dynamics (6.3) according to Definition 6.1. Introducing the Minkowski sum $\oplus$, the preset is [66]

$$\mathcal{O}(\Omega) = \Big\{x \in \mathbb{R}^2 \mid \exists u \in \mathcal{U}_s \ : \ x + T_s u \in \Omega\Big\} \tag{6.14a}$$

$$= \Big\{x \in \mathbb{R}^2 \mid \exists y \in \Omega, \ \exists u \in \mathcal{U}_s \ : \ x = y - T_s u\Big\} \tag{6.14b}$$

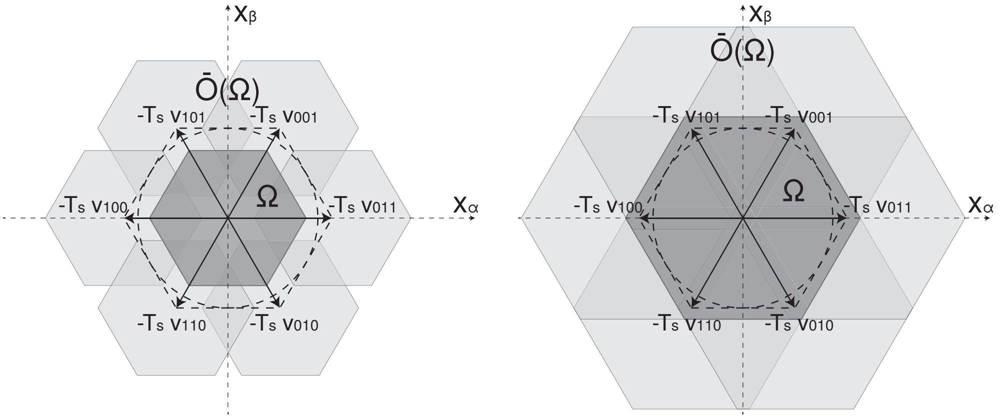
$$= \Omega \oplus (-T_s \mathcal{U}_s). \tag{6.14c}$$

A useful interpretation of the Minkowski sum is that it is the union of all translated copies of Ω by the vectors of $(-T_s\mathcal{U}_s)$ [66]

$$\mathcal{O}(\Omega) = \bigcup_{u \in \mathcal{U}_s} (\Omega - T_s u) = T_s \bar{u} + \bigcup_{v_{\alpha\beta} \in \mathcal{V}_s} (\Omega - T_s v_{\alpha\beta}). \tag{6.15}$$

Thus, the preset is the union of $2^3 = 8$ polytopes, which is shown in Figure 6.3.

Properties of the preset can be derived analyzing the set $\bar{\mathcal{O}}(\Omega) \overset{\text{def}}{=} \mathcal{O}(\Omega) - T_s\bar{u} = \bigcup_{v_{\alpha\beta} \in \mathcal{V}_s} (\Omega - T_s v_{\alpha\beta})$, which is shown in Figure 6.4. The properties of this set depend on the size of Ω with respect to the length of the vectors $T_s v_{\iota\iota\iota}$, i.e. X_l with respect to the rated voltage v_r. If the length of the side of Ω (the length of the side of the polytopes forming $\bar{\mathcal{O}}(\Omega)$) is smaller than the distance of the centers of two adjacent polytopes of $\bar{\mathcal{O}}(\Omega)$, then $\bar{\mathcal{O}}(\Omega)$ is nonconvex. This is the case if Ω is smaller than $T_s \operatorname{hull} \mathcal{V}_s = T_s \mathcal{V}_d$, i.e. $X_l < T_s v_r$ and it is portrayed in Figure 6.4(a). If X_l is further reduced, then the polytopes forming $\mathcal{O}(\Omega)$ are not even connected.



(a) Ω with $X_l < T_s v_r$

(b) Ω with $X_l \geq T_s v_r$

Figure 6.4.: The set $\bar{\mathcal{O}}(\Omega)$

If Ω is larger or equivalent than $T_s\mathcal{V}_d$, i.e. $X_l \geq T_s v_r$, then adjacent polytopes forming $\bar{\mathcal{O}}(\Omega)$ overlap such that the resulting $\bar{\mathcal{O}}(\Omega)$ is convex, which is shown in Figure 6.4(b). This set is described by

$$\bar{\mathcal{O}}(\Omega) = \left\{ x \in \mathbb{R}^2 \mid \mathbf{H}x \leq X_l + T_s v_r \right\} = \Omega \oplus (-T_s\mathcal{V}_d). \tag{6.16}$$

This simplification is useful to obtain the following result.

Proposition 6.1. *Let $\bar{u} \in \operatorname{int}\mathcal{V}_d$ and let the set Ω be defined by $X_l \geq T_s v_r$, then*

$$\Omega \subset \operatorname{int}\mathcal{O}(\Omega). \tag{6.17}$$

Proof. First, the largest set $\bar{\Omega}$ is defined such that

$$\bar{\Omega} \subset \mathcal{O}(\Omega) = \bar{\mathcal{O}}(\Omega) + T_s\bar{u} \quad \forall \bar{u} \in \operatorname{int}\mathcal{V}_d. \tag{6.18}$$

This set is provided by the intersection of all translated copies of $\bar{\mathcal{O}}(\Omega)$ by the vectors of $T_s\mathcal{V}_d$

$$\bar{\Omega} = \bigcap_{\bar{u} \in \operatorname{int}\mathcal{V}_d} (\bar{\mathcal{O}}(\Omega) + T_s\bar{u}), \tag{6.19}$$

which is known as the Pontryagin (Minkowski) difference [66]

$$\bar{\Omega} = \bar{\mathcal{O}}(\Omega) \ominus (-T_s \operatorname{int}\mathcal{V}_d) = \left\{ y \in \mathbb{R}^2 \mid y - T_s\bar{u} \in \bar{\mathcal{O}}(\Omega)\ \forall \bar{u} \in \operatorname{int}\mathcal{V}_d \right\}. \tag{6.20}$$

If $X_l \geq T_s v_r$, then the set $\bar{\mathcal{O}}(\Omega)$ can be written according to (6.16), that is

$$\bar{\Omega} = \Omega \oplus (-T_s \mathcal{V}_d) \ominus (-T_s \operatorname{int} \mathcal{V}_d). \tag{6.21}$$

This result is interpreted as follows. For any $\bar{u} \in \operatorname{int} \mathcal{V}_d$, a sufficiently small positive scalar $b \in \mathbb{R}_{>0}$ can be found such that $\bar{u} \in \{x \in \mathbb{R}^2 \mid \mathbf{H}x \leq v_r - b\}$, that yields

$$\bar{\Omega} = \left\{x \in \mathbb{R}^2 \mid \mathbf{H}x \leq X_l + T_s v_r - T_s(v_r - b) = X_l + T_s b\right\}. \tag{6.22}$$

Since $b > 0$ the set $\Omega \subset \operatorname{int} \bar{\Omega}$ and since $\bar{\Omega} \subset \mathcal{O}(\Omega)$ the set

$$\Omega \subset \operatorname{int} \mathcal{O}(\Omega). \tag{6.23}$$

□

Thus, any sublevel set Ω defined by $X_l \geq T_s v_r$ belongs to the interior of its preset $\mathcal{O}(\Omega)$ if $\bar{u} \in \operatorname{int} \mathcal{V}_d$. If Ω is defined by $X_l < T_s v_r$, then $\Omega \not\subset \operatorname{int} \mathcal{O}(\Omega)$ in general. An example of this case is shown in Figure 6.3(a).

6.2.3. Set Stabilizability

The Proposition 6.1 has the following interesting consequence.

Proposition 6.2. *Let $\bar{u} \in \operatorname{int} \mathcal{V}_d$ then*

$$\exists u \in \mathcal{U}_s \ : \ \begin{cases} \Gamma(x + T_s u) - \Gamma(x) < 0 & \textit{if } x \in \mathbb{R}^2 \setminus T_s \mathcal{V}_d, \\ \Gamma(x + T_s u) - T_s v_r \leq 0 & \textit{otherwise.} \end{cases} \tag{6.24}$$

Proof. If $\bar{u} \in \operatorname{int} \mathcal{V}_d$ and $x \in \mathbb{R}^2 \setminus T_s \mathcal{V}_d$, a sublevel set Ω and its preset $\mathcal{O}(\Omega)$, where $\Omega \subset \operatorname{int} \mathcal{O}(\Omega)$ by Proposition 6.1, can be found such that $x \in \mathcal{O}(\Omega) \setminus \Omega$. Moreover, an input $u \in \mathcal{U}_s$ exists such that $x + T_s u \in \Omega$ by the Definition 6.1. Since $\Gamma(y) < \Gamma(x)$ for all $y \in \Omega$ and $x \in \mathcal{O}(\Omega) \setminus \Omega$, the following equation is obtained

$$\exists u \in \mathcal{U}_s \ : \ \Delta\Gamma(x, u) = \Gamma(x + T_s u) - \Gamma(x) < 0 \ \ \forall x \in \mathbb{R}^2 \setminus T_s \mathcal{V}_d. \tag{6.25}$$

Moreover, $T_s \mathcal{V}_d \subseteq \mathcal{O}(T_s \mathcal{V}_d)$, i.e. $T_s \mathcal{V}_d$ is said to be control invariant [19]. Thus, an $u \in \mathcal{U}_s$ can be found such that $x + T_s u \in T_s \mathcal{V}_d$ for all $x \in T_s \mathcal{V}_d$, which yields

$$\exists u \in \mathcal{U}_s \ : \ \Gamma(x + T_s u) - T_s v_r \leq 0 \ \ \forall x \in T_s \mathcal{V}_d. \tag{6.26}$$

□

According to Proposition 6.2, the value of a candidate control Lyapunov function can be decreased for all $x \in \mathbb{R}^2 \setminus T_s\mathcal{V}_d$. Thus, any state $x \in \mathbb{R}^2 \setminus T_s\mathcal{V}_d$ can be steered towards the set $T_s\mathcal{V}_d$. Moreover, for all $x \in T_s\mathcal{V}_d$ there exists an $u \in \mathcal{U}_s$ such that $x + T_s u \in T_s\mathcal{V}_d$. As a consequence, there exists a sequence $u[k], u[k+1], \ldots$ such that

$$\lim_{k\to\infty} x[k] \in \mathcal{D} \overset{\text{def}}{=} T_s\mathcal{V}_d. \tag{6.27}$$

6.3. CFTOC and RHC

So far it has been established that there exists always an $u \in \mathcal{V}_s$ such that any $x \in \mathbb{R}^2$ can be steered towards the set $x \in T_s\mathcal{V}_d$, which is named *terminal set*. Thus, the control goals depend on whether the state lies within or without the terminal set. If $x \in \mathbb{R}^2 \setminus T_s\mathcal{V}_d$, there exist an input $u \in \mathcal{V}_s$ such that the state can be driven closer (in the sense of reducing a candidate control Lyapunov function value) towards the terminal set. This operation is interpreted as *dynamic*, where the main focus is reducing the control error fast. Once state has reached the terminal set, i.e. $x \in T_s\mathcal{V}_d$, the state cannot be steered further towards the origin in general. This operation is interpreted as *steady-state*, where the state has to be kept within the terminal set. This goal should be achieved producing small energy losses due to switching.

In this section, the Lyapunov-based MPC is designed in analogy to the CCS-MPC controller, which yields the CFTOC

$$\underset{u_0,\ldots,u_{N-1}}{\text{minimize}}\ J(x_0, u_0, \ldots, u_{N-1}) \tag{6.28a}$$

$$\text{subject to } x_{j+1} = x_j + T_s u_j; \tag{6.28b}$$

$$u_j \in \mathcal{U}_{s,j} \overset{\text{def}}{=} \mathcal{V}_s - \bar{u}_j; \tag{6.28c}$$

$$\begin{cases} \Gamma(x_j + T_s u_j) \leq T_s v_r & \text{if } x_j \in T_s\mathcal{V}_d; \\ \Gamma(x_j + T_s u_j) < \Gamma(x_j) & \text{otherwise.} \end{cases} \tag{6.28d}$$

The dynamic operation CFTOC is defined by a cost function (6.28a) and constraints. The constraint (6.28b) introduces the plant dynamics and (6.28c) introduces the FCS input constraint. The main difference compared with CCS-MPC is provided by the Lyapunov stability criteria (6.28d). In FCS-MPC, the candidate control Lyapunov function is required to be strictly decreasing for all future states $x_j \in \mathbb{R}^2 \setminus T_s\mathcal{V}_d$ and to be smaller or equal $T_s v_r$ (candidate control Lyapunov function value defining the terminal set) otherwise. Regarding set stability it is sufficient to apply this constraint only to the first predicted step. However, requiring it also for future steps comes along with computational advantages since a significant amount of sequences can be declared as infeasible and do not need to be computed.

Clearly, the CFTOC (6.28) is solved by an input U and state X sequence. A sequence is said to be *feasible* if it satisfies the constraints. Moreover, it is said to be *optimal* if it is feasible and provides the lowest cost of all feasible sequences.

Proposition 6.3. *If $\bar{u}_j \in \operatorname{int} \mathcal{V}_d$, then the CFTOC (6.28) is feasible.*

Proof. Feasibility is given if there exists at least one sequence which satisfies (6.28b), (6.28c), (6.28d). By Proposition 6.2, a $u_j \in \mathcal{U}_{s,j}$ does always exist such that $\Gamma(x_j + T_s u_j) \leq T_s v_r$ if $x_j \in T_s \mathcal{V}_d$ and $\Gamma(x_j + T_s u_j) < \Gamma(x_j)$ otherwise. Thus, a sequence $\mathrm{U} = [u_0', \ldots, u_{N-1}']'$ can be constructed, which is feasible and yields a feasible state sequence X (any X is feasible). □

Similar to CCS-MPC, the FCS-MPC uses the receding horizon policy provided by Algorithm 5.1 to implement an input. In other words, the optimal input $\mathrm{U}^\star$ and state $\mathrm{X}^\star$ sequence is computed and the first optimal input is applied to the plant $u[k] = u_0^\star$. Thus, the following can be said regarding set stability.

Theorem 6.1. *If $\bar{u}_j \in \operatorname{int} \mathcal{V}_d$ for all future time steps, then the MPC using the receding horizon policy Algorithm 5.1 solving the CFTOC (6.28) at each time step is set stable, i.e. the state x converges to the set $\mathcal{D} = T_s \mathcal{V}_d$ and remains within $\mathcal{D}$ for all future time.*

Proof. If $\bar{u}_j \in \operatorname{int} \mathcal{V}_d$ for all future time steps, then the CFTOC (6.28) is feasible for all time. Then, the constraint (6.28d) ensures that x converges towards the set $\mathcal{D} = T_s \mathcal{V}_d$ and remains within $\mathcal{D}$ for all future time according to Proposition 6.2. □

6.4. Cost Function

Now, a cost function is chosen for the CFTOC (6.28). Although the CFTOC ensures convergence to the terminal set, the cost function should support the fast convergence. Moreover, the cost function influences the number of switching transitions[2], which are used to converge to the terminal set and to stay within the terminal set. Other control goals can be added to the cost function, e.g. each leg can be forced to

[2] The number of switching transitions, i.e. the switching frequency, is proportional to the switching losses, see Chapter 3.

achieve similar number of switching transitions. However, such goals are omitted for simplicity.

Thus, the cost function is defined as the sum of two components

$$J(x_0, u_0, \dots, u_{N-1}) = J_1(x_0, u_0, \dots, u_{N-1}) + \gamma J_2(x_0, u_0, \dots, u_{N-1}) \tag{6.29}$$

The first cost component $J_1(.)$ adds a Lyapunov related component to the cost. Clearly, the Lyapunov function can always be reduced if the state lays outside the terminal set terminal set. Thus, a penalty function [59], [81], [82] styled approach is introduced. The cost component is defined to be zero if within the terminal set and the Lyapunov function otherwise. To provide continuity, the Lyapunov level $T_s v_r$ (limit of the terminal set) is subtracted from the Lyapunov function, which

Algorithm 6.2: Recursive algorithm to solve CFTOC (6.28) by full enumeration

```
if $\|\bar{u}_j\| < v_r$ then
    /* $\bar{u}_j \in \operatorname{int}\mathcal{V} \subset \operatorname{int}\mathcal{V}_d$; (6.28) has a solution */
    for $s_{ph,j} \leftarrow [0,0,0]'$ to $[1,1,1]'$ do
        /* for every vector $s_{ph,j}$ yielding $u_j$, do */
        $x_{j+1} \leftarrow x_j + T_s u_j$                /* compute next state using $s_{ph,j}$ */
        if $j < N-1$ then
            /* Recursive function call:  compute future steps; return */
            /* (i) optimal input sequence $u^\star_{j+1}, \dots, u^\star_{N-1}$ */
            /* (ii) the cost $J^\star_{s_{ph,j},j+1\to N}$ */
        else
            $J^\star_{s_{ph,j},j+1\to N} \to 0$
        /* compute cost of sequence $j$ to $N$ for vector $s_{ph,j}$ */
        $J_{s_{ph,j},j\to N} \leftarrow J_{s_{ph,j},j} + J^\star_{s_{ph,j},j+1\to N}$
        if $J_{s_{ph,j},j\to N} < J^\star_{j\to N}$ then
            /* Pick the vector with minimum cost and store it */
            $J^\star_{j\to N} \leftarrow J_{s_{ph,j},j\to N}$
            $s^\star_{ph,j} = s_{ph,j}$
    if $j = 0$ then
        $v^\star_{\alpha\beta} \leftarrow u^\star_0 - \bar{u}_0$                /* apply first optimal input */
else
    /* $\bar{u} \notin \operatorname{int}\mathcal{V} \subset \operatorname{int}\mathcal{V}_d$; (6.28) may not have a solution */
    stop
```

yields

$$J_1(.) \overset{\text{def}}{=} \sum_{j=1}^{N} \begin{cases} 0 & \text{if } x_j \in T_s\mathcal{V}_d; \\ \Gamma(x_j) - T_s v_r & \text{otherwise.} \end{cases}. \tag{6.30}$$

The second component $J_2(.)$ penalizes the number of switching transients used to achieve the control goal adding the cost function component

$$J_2(.) \overset{\text{def}}{=} \sum_{j=0}^{N-1} \|s_{ph,j} - s_{ph,j-1}\|_1. \tag{6.31}$$

The switching penalization is applied globally. The weighting factor γ defines the importance of the cost function components with respect to each other. To ensure fast convergence γ is chosen to be small. Thus, the switching penalization contributes primarily when the state is within the terminal set, where $J_1(.)$ is zero.

6.5. Implementation

FCS-MPC requires to solve the CFTOC (6.28), which is a mixed-integer programming problem due to the constraint $u_j \in \mathcal{U}_{s,j}$. The conceptually simplest approach to solve this problem is to simply enumerate all input possibilities and compute the feasible optimal input sequence, e.g. using the recursive Algorithm 6.2

However, this approach is computationally suboptimal and can be improved. First, one of the passive vectors is neglected according to Proposition 3.1. The passive

Table 6.1.: Number of cost function evaluations

Horizon N	**Full Enum.**	**Opt. Enum.**	
	(8^N evaluations)	MEAN	MAX
1	8	1.4	4
2	64	3.9	12
3	512	7.1	40
4	4096	11.2	113
5	32768	16.9	261
6	262144	25.0	666
7	2097152	36.7	710
8	16777216	53.2	762

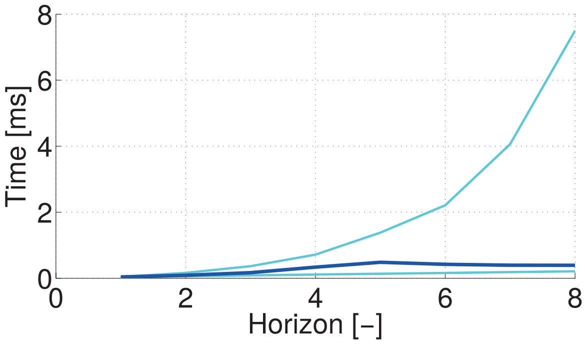


Figure 6.5.: FCS-MPC execution time (mean, minimum, and maximum)

vector to neglect can be chosen independent of the future evolution of the switching sequence and therefore computationally cheap. Then, some of the sequences are infeasible since they violate the Lyapunov constraint. Thus, they can be safely neglected (this approach is similar to [36] where some sequences violate a hysteresis bound).

Moreover, the approach can be further improved using *branch-and-bound*, i.e. *branch-and-cut* techniques. Once a full input sequence has been evaluated, its cost is an upper bound of the optimal cost. This upper bound is updated if another sequence provides an inferior cost. In other words, the minimum cost, which has been found so far by the algorithm, is stored. Any sequence which exceeds this cost can be safely discarded even before evaluating the full sequence. Using a combination of these methods, the number of cost function evaluations can be significantly reduced. A comparison of full enumeration (full enum.) and optimized enumeration (opt. enum.) is shown in Table 6.1.

It is possible to further optimize the algorithm using *branch-and-bound* theory. For example, the presented strategies are exact in the sense that they find the global optimal sequence, which is considered to be computationally expensive. Thus, mixed integer programming algorithms typically use a heuristic, which can declare a sequence to be (approximately) optimal if it is “close enough” to the exact optimum. However, the enumeration algorithm is not further optimized in this text. In a real time embedded environment, a CFTOC problem with a few prediction steps N can already be computationally critical. Thus, the scaling of the computational complexity of the CFTOC with long prediction horizon is considered to be of secondary importance.

Now, the execution time of the FCS-MPC is reviewed. The Algorithm 6.2 with optimizations is timed on the experimental test bench outlined in Appendix D.

The minimum, mean, and maximum execution time is $33.80\mu s$, $40.21\mu s$, $43.23\mu s$, respectively, using the prediction horizon $N = 1$. Thus, the Algorithm 6.2 is more expensive to compute than the nonlinear controller and CCS-MPC, which are shown in Table 5.1 in Chapter 5. However, the computation requirements of FCS-MPC are still sustainable on most systems. The execution time using the prediction horizon $N = \{1, \dots, 8\}$ are depicted in Figure 6.5. The graph shows that the minimum and mean execution time scales reasonably well with the prediction horizon. However, the worst case timing still grows exponentially. Thus, further development, e.g. introducing heuristics, is required if a long prediction horizon is desired.

6.6. Dynamic Operation

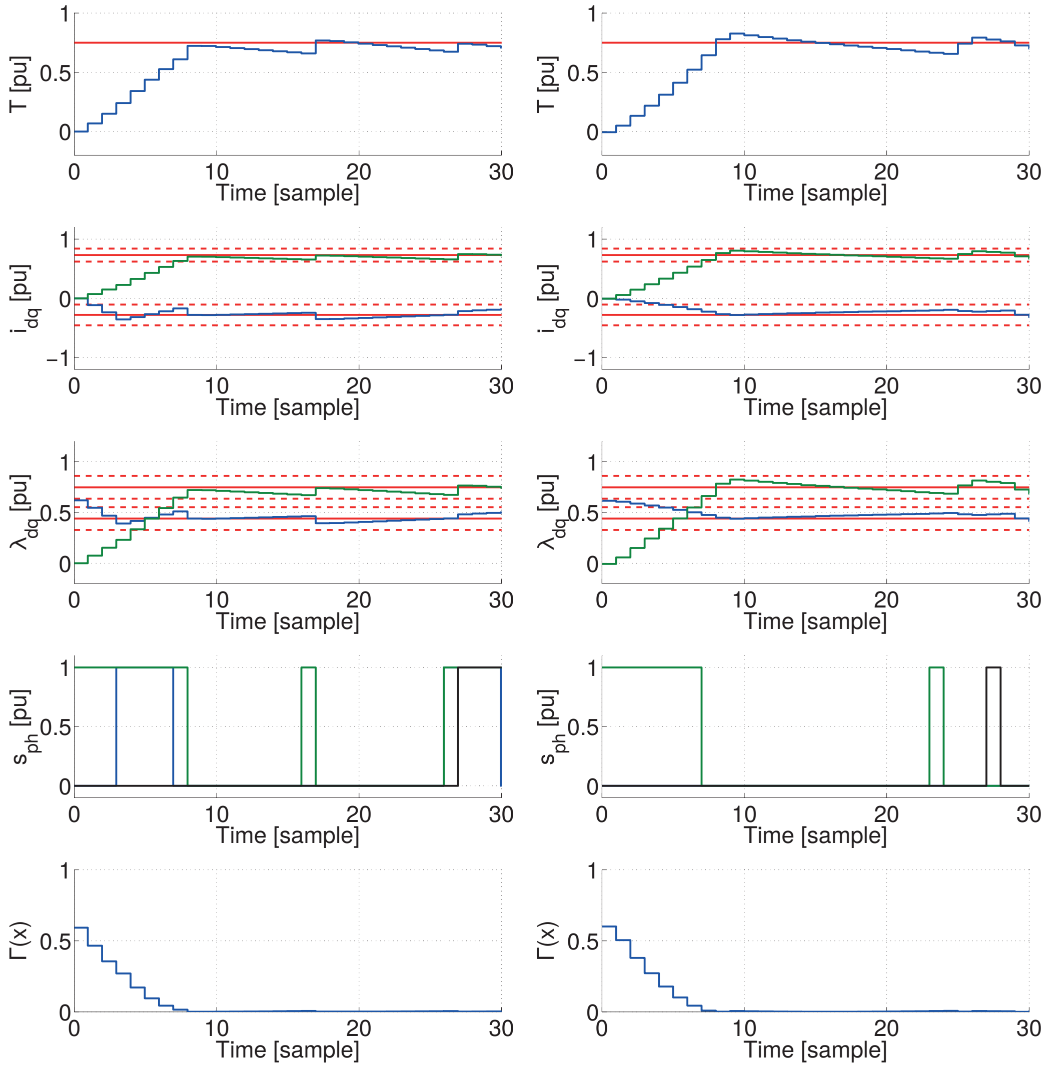
In this section, the proposed FCS-MPC with prediction horizon $N = 1$ and optimal reference generation procedure is reviewed in dynamic operation. The case $N = 1$ is chosen since it provides the minimum implementation. Moreover, a controller with longer horizon tends to perform better since optimality with respect to a longer horizon is achieved and a controller tends to infinite-time MPC extending the horizon [19]. First, a torque reference step is applied to the reference generation procedure. This results in both, a d and q current, i.e. flux, reference step. The step is applied at standstill $\omega = 0$ and during the period of observation, the machine speed remains approximately zero. The simulation and experimental results are shown in Figure 6.6. From top, the figure shows the reference (R, red) and actual (T, blue) torque; the reference current vector (r_{dq}, solid red), the actual d (i_d, blue) and q (i_q, green) current; the reference flux vector ($\bar{r}_{dq}$, solid red), the actual d (λ_d, blue) and q (λ_q, green) flux; the duty cycle d_{ph}; and the Lyapunov function $\Gamma(x)$ with $x = \lambda_{\alpha\beta} - \bar{r}_{\alpha\beta}$. FCS-MPC cannot converge to the origin, i.e. to the reference flux and current vector, but it will converge to and remain in the set $\mathcal{D} = T_s \mathcal{V}_d$. In the dq reference frame, the bounds provided by $\mathcal{D}$ are time varying. Figure 6.6 shows the constant but worst case flux bounds (red, dashed), which are computed as

$$\bar{r}_{dq,bound} = \bar{r}_{dq} \pm \frac{2}{3} T_s v_c, \tag{6.32}$$

and the worst case current bounds (red, dashed) obtained via transformation

$$r_{dq,bound} = \mathbf{L}^{-1} \left(\bar{r}_{dq,bound} - \psi_{dq} \right). \tag{6.33}$$

The results confirm the FCS-MPC design and implementation. They show that the Lyapunov function is strictly decreasing once the reference step is applied. The controller applies a voltage according to the input constraint $\mathcal{U}_s$ until the flux, i.e.



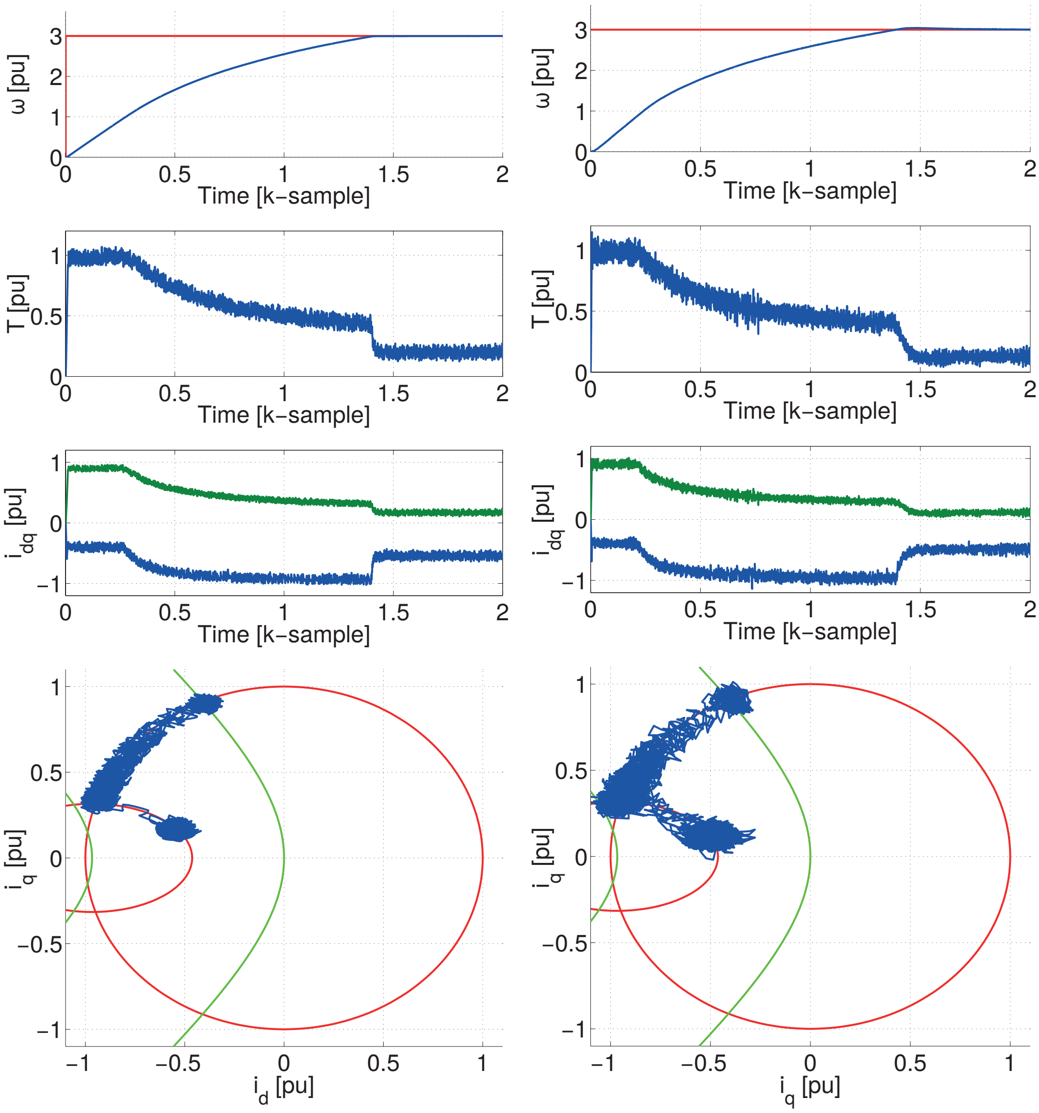
(a) Simulation

(b) Experimentation

Figure 6.6.: Torque step from $0pu$ to $0.75pu$ using FCS-MPC with horizon $N = 1$

current, enters the set $\mathcal{D}$. Clearly, the available voltage vectors depend on the electrical angle and as a consequence also the raise time does.

Moreover, a speed reference step from standstill to $3pu$ is evaluated using an anti-windup proportional-integral speed controller. Since the reference step is large, the speed loop saturates and the maximum available torque is applied approximately

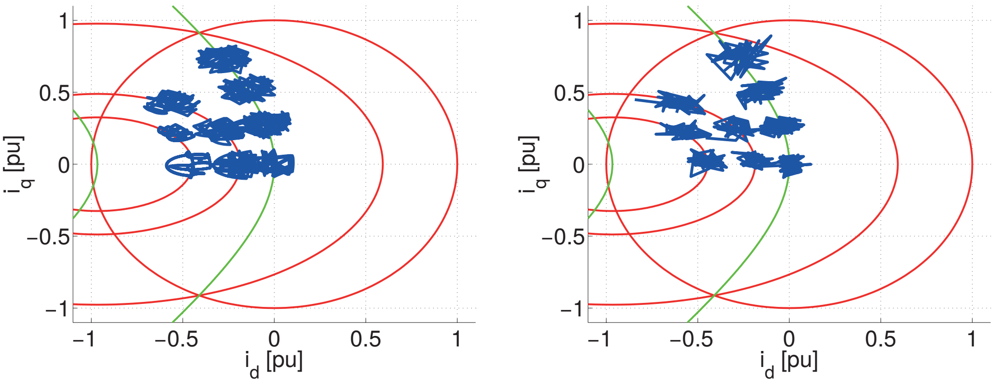


(a) Simulation

(b) Experimentation

Figure 6.7.: Speed step form standstill to $3pu$ using FCS-MPC with horizon $N = 1$

until the speed reference is reached. The results are presented in Figure 6.7. From top, the figure shows the reference (red) and actual (ω, blue) speed; the torque (T); the actual d (i_d, blue) and q (i_q, green) current; and the current i_{dq} in the current state-plane.



(a) Simulation

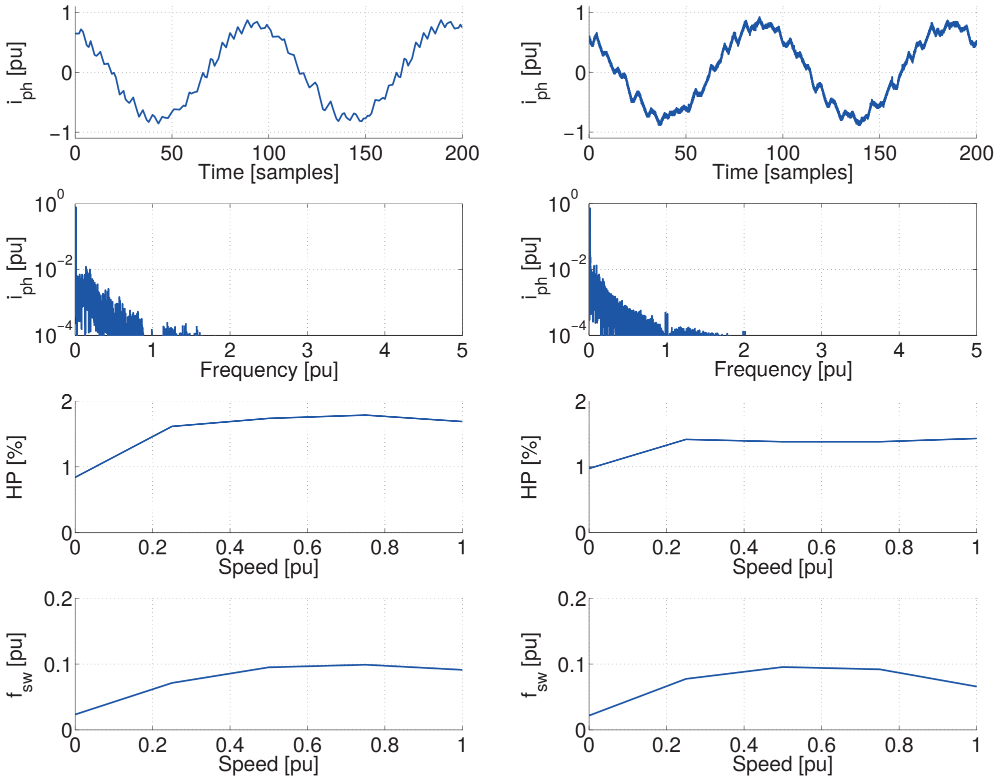
(b) Experimentation

Figure 6.8.: Steady-state operation points using CCS-MPC with horizon $N = 1$ and reference generation procedure

The results confirm the design and implementation of the FCS-MPC and the reference generation procedure. At low speed, the maximum torque and acceleration are constant. The current producing maximum torque lies on the intersection of the MTPA and the isocurrent locus. Increasing the speed beyond rated speed, this operation point is not available anymore due to the voltage limit. Thus, the current moves along the isocurrent locus until the reference speed is achieved. Then the current moves along the isoflux locus to the operation point which provides the torque to maintain the machine at reference speed. FCS-MPC achieves a significantly higher ripple than nonlinear control and CCS-MPC since it does not converge to the origin. On the other hand it achieves also a significantly smaller (and variable) switching frequency and related power losses.

6.7. Steady-State Operation

In this section, the proposed FCS-MPC with prediction horizon $N = 1$ and optimal reference generation procedure is reviewed in steady-state operation. First, different load torques are applied such that the machine produces a constant torque T at different speeds ω. The following cases are evaluated: $\omega = 1.0pu$ with $T = \{0pu, 0.25pu, 0.5pu, 0.75pu\}$; $\omega = 2.0pu$ with $T = \{0pu, 0.25pu, 0.5pu\}$; and $\omega = 3.0pu$ with $T = \{0pu, 0.25pu\}$. The simulation and experimental results are shown in Figure 6.8. Ideally, steady-state operation with given conditions (torque, speed) would lead to a single point on the state plane. In practice, the states stay in the neighborhood of this point due to power converter switching, measurement noise,



(a) Simulation

(b) Experimentation

Figure 6.9.: Current quality using FCS-MPC with horizon $N = 1$

etc. The results show that the drive system applies states according to the reference generation procedure. The controller operates on the MTPA trajectory at $\omega = 1.0pu$. At $\omega = 2.0pu$ and $\omega = 3.0pu$, the system operates on the isoflux locus, which is defined by the speed ω.

Finally, the steady-state current quality, which is achieved by the FCS-MPC with horizon $N = 1$, is evaluated. The results are shown in Figure 6.9. From top, the typical current waveform and spectrum is presented (at $0.75pu$ speed and $0.75pu$ torque). In these figures, frequencies are normalized with the sampling frequency $f_s = 1/T_s$. Moreover,the harmonic power (HP) and the average switching frequency f_{sw} is evaluated as a function of the machine speed.

In this section, the focus lies on harmonics, which are introduced by control and switching. Thus, the performance index *Harmonic Power* (HP), which is defined by (4.27), is used. It captures the harmonics from interest evaluating the frequency band from $0.1f_s$ to $10f_s$. The index is proportional to the power of the harmonics in the defined harmonic band as average of the three phases. The switching frequency is evaluated counting the number of switch transitions per phase $n_{ph} \in \mathbb{N}^3$ in an evaluation interval $T_{sw} \in \mathbb{R}_+$ and the average switching frequency is computed according to (4.29).

The FCS-MPC current quality analysis is presented to show some fundamental properties of the control concept. In contrast to using a modulation scheme, FCS-MPC does not lead to emphasized switching harmonics but to accumulations of harmonics. Moreover, the switching frequency is significantly lower than the sampling frequency and it depends on the machine speed. The switching properties of FCS-MPC make the concept interesting for applications where sufficiently high sampling frequency is desired keeping the switching frequency low, e.g. in high power or servo drives as alternative to hysteresis based approaches [43], [97]. With respect to acoustic noise, FCS-MPC has some inherent advantages over controllers PWM since it does not generate emphasized switching harmonics. Thus, particular tones are missing in the acoustic spectrum. On a downside, the random switching properties make it difficult to obtain realistic upper bounds on the average switching frequency.

Part III.

Conclusion

Contributions

This thesis focuses on advanced torque control of permanent magnet synchronous motor drives. A novel modular **control structure** is introduced to simplify the control design and implementation. The *control framework* contains static mappings to simplify the dynamic control problem.

- The *reference generation* transforms a reference torque value into a reference state vector. The transformation is optimized for high efficiency using the *Maximum Torque per Ampere* (MTPA) criteria below rated speed and in *field weakening* operation.
- The *actuation scheme* considers actuating a control input, i.e. terminal voltage, to the gates of the power electronic converter and defines the input constraint set. A *Convex Control Set* (CCS) is obtained using duty cycles, which are actuated to the inverter using PWM. *Direct actuation* means actuating switch states, which yields the *Finite Control Set* (FCS).

Dynamic control considers the evolution of the control error over time.

- The controller is written in the $\alpha\beta$ *virtual flux* space. The flux stat-space model is simpler than its current equivalent. The $\alpha\beta$ reference frame yields a time invariant input constraint.
- A change of states transforms the reference tracking problem into a regulation problem. The resulting control problem is solved by a linear combination of a *feedforward* and *feedback* controller.
- The feedback controller is designed as *Model Predictive Control* (MPC) solving a *Constrained Finite Time Optimal Control* (CFTOC) problem at each sampling time instant. The control goal is defined as *cost function* and the open-loop behavior is optimized over N future time steps called *prediction horizon*.

A **stability and feasibility analysis** is carried out for CCS-MPC and FCS-MPC.

- The *Lyapunov-based* MPC approach is introduced to drive systems. A *Control Lyapunov Function* (CLF) constraint is introduced to the CFTOC, which requires that at least the first of the optimal input sequence decreases the value of a Lyapunov function.
- The CLF is designed for the constraint system and it is shown that the resulting CFTOC is persistently feasible (feasible for all time).

- The switching behavior is transparent to CCS-MPC. Thus, the error can be steered to the origin and the closed-loop system is shown to be stable in the sense of Lyapunov.
- FCS-MPC considers the switching and cannot arbitrarily reduce the control error, i.e. value of a Lyapunov function. It is shown that FCS-MPC can always steer the control error to a neighborhood of the origin, where it remains ultimately bounded. Thus, it is shown to be set-stable.

Efficient implementations to solve the CFTOC are developed.

- Code is typically executed on embedded hardware with limited computation capabilities and needs to provide a result, i.e. an updated input, within a (small) sampling period. Thus, computationally efficient algorithms are required.
- The CCS-MPC CFTOC is a (convex) linear or quadratic programming problem (neglecting CLF, which lead to quadratic constraints). With horizon $N = 1$, the proposed CCS-MPC is shown to be solvable by an analytic algorithm, which yields the exact solution to the CFTOC efficiently.
- The FCS-MPC CFTOC is a mixed-integer linear or quadratic programming problem and consequently more demanding to solve. *Branch-and-bound*, i.e. *branch-and-cut*, techniques are added to the typical enumeration strategy to improve the computational efficiency.

An *experimental evaluation* is carried out for the proposed concepts.

- The control code is developed on a Software-in-the-Loop (SiL) platform and the same code is evaluated on an experimental test bench in dynamic operation (speed and torque reference step) and in steady-state conditions (location of operation points and current quality).
- CCS-MPC and FCS-MPC achieve similarly good dynamic results and are able to work on the desired operation points. However, they differ significantly in the switching behavior. Differences include a constant or variable switching frequency, sampling to switching frequency ratio, current quality, and harmonic spectrum (with related acoustic noise).

Future Work

In this text, an advanced control system for PMSM drive systems has been developed. The structure is modular such that parts of the control system can be improved individually.

- The reference generation procedure can be further improved either to take nonlinear effects into account or to reduce the computation complexity to solve the problem. Multidimensional lookup tables can be introduced to solve the problem and/or subproblems.
- MPC in the virtual flux space can be combined with nonlinear transformations to take the magnetic saturation and cross-saturation into account. Both theoretical and practical advantages of such an approach need to be studied.
- The fast gradient method can be combined with CCS-MPC in the virtual flux space to solve a CFTOC with horizon $N > 1$. The numeric solver is advantageous for input-constrained MPC CFTOC, which are convex.

Part IV.

Appendix

Appendix A.

Clarke and Park Transformation

Transformations are commonly used to simplify the analysis and modeling of three-phase systems [1], [14]. Popular transformations are the Clarke or $\alpha\beta$ transformation [31] and the Park or dq transformation [74]. These methods are particularly interesting if a system is neutral point isolated. There, the dimension of a variable, e.g. the voltage, current, or flux, can be reduced. In this chapter, the transformations are treated for a general three-phase vector $x_{ph}(t) = [x_1(t), x_2(t), x_3(t)]' \in \mathbb{R}^3$, where $.(t)$ denotes the dependence on time $t \in \mathbb{R}_+$.

A.1. Full Transformation

A three-phase entity can be transformed into the $\alpha\beta 0$ reference frame using the transformation

$$\mathbf{T}_{\alpha\beta 0} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \tag{A.1}$$

that yields the vector $x_{\alpha\beta 0}(t) = \mathbf{T}_{\alpha\beta 0} x_{ph}(t) \in \mathbb{R}^3$.

The transformation is invertible and can be applied to systems with arbitrary waveforms without loss of generality. It is most interesting if a three-phase vector $x_{ph}(t)$ has, or is desired to have, components with approximately sinusoidal waveforms (sinusoidal with a non-dominant amount of harmonics) and the system is approximately balanced (each wave has a similar magnitude and the signals are approximately shifted by the angle $2\pi/3$). In this case, the transformed vector $x_{\alpha\beta 0}(t)$ rotates in the plane spanned by the first two coordinates (the $\alpha\beta$ plane) with angular velocity $\omega(t)$ and the 0 axis component is typically small.

Moreover, an $\alpha\beta 0$ vector can be transformed into the $dq0$ reference frame using the transformation

$$\mathbf{T}_{dq0}(\epsilon(t)) = \begin{bmatrix} \cos\epsilon(t) & \sin\epsilon(t) & 0 \\ -\sin\epsilon(t) & \cos\epsilon(t) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{A.2}$$

that yields the vector $x_{dq0}(t) = \mathbf{T}_{dq0}(\epsilon(t))x_{\alpha\beta 0}(t) \in \mathbb{R}^3$.

Also this transformation is invertible and can be applied to systems with arbitrary waveforms without loss of generality. This reference frame rotates with velocity $\omega(t)$ in the plane spanned by the first two coordinates (the dq plane) with respect to the zero axis. It is defined by the transformation angle $\epsilon(t)$ with $\dot{\epsilon}(t) = d\epsilon(t)/dt = \omega(t)$. The angle is typically aligned with an external reference (voltage source or electric motion force) such that system equations simplify. This transformation is interesting since it transforms a rotating vector, which is defined by AC $\alpha\beta$ components, into a stationary vector, which is defined by DC dq components.

A.2. Partial Transformation

Currents with zero component cannot circulate in neutral point isolated three-phase systems. The law of Kirchhoff yields $i_0(t) = \frac{1}{3}\mathbf{1}'i_{ph}(t) = 0$, that defines the nullspace $\mathbf{1}'$, i.e. $\mathbf{1}'i_{ph}(t) \equiv 0$. In most cases, it is still possible to apply voltages with zero component $v_0(t) = \frac{1}{3}\mathbf{1}'v_{ph}(t) \neq 0$ but $v_0(t)$ faces an open circuit and does not drive currents in the system. Thus, it is common practice to omit the nullspace $\mathbf{1}'$ to further simplify the treatment of such systems. This approach is safe for neutral point isolated three-phase systems.

A three-phase entity can be transformed into the $\alpha\beta$ reference frame using the transformation

$$\mathbf{T}_{\alpha\beta} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}, \tag{A.3}$$

that defines the vector $x_{\alpha\beta}(t) = \mathbf{T}_{\alpha\beta}x_{ph}(t) \in \mathbb{R}^2$.

This transformation cannot be inverted since information on the zero component is lost in general. However, $\mathbf{1}'$ is a nullspace such that $x_0 = 0$ for currents (or fluxes). In this case, back-transformation is possible using $x_{ph}(t) = \mathbf{T}^{\dagger}_{\alpha\beta}x_{\alpha\beta}(t)$, where $.^{\dagger}$ denotes the Moore-Penrose pseudoinverse [77]. For voltages, x_0 is often undefined and can be chosen to the advantage of the system. Such a strategy to transform $x_{\alpha\beta}$ into x_{ph} is called modulation. The $\alpha\beta$ reference frame spans a plane where an approximately balanced sinusoidal three-phase entity results in a rotating $\alpha\beta$ vector.

Moreover, an $\alpha\beta$ vector can be transformed into the dq reference frame using the transformation

$$\mathbf{T}_{dq}(\epsilon(t)) = \begin{bmatrix} \cos\epsilon(t) & \sin\epsilon(t) \\ -\sin\epsilon(t) & \cos\epsilon(t) \end{bmatrix}, \tag{A.4}$$

which defines the vector $x_{dq}(t) = \mathbf{T}_{dq}(\epsilon(t))x_{\alpha\beta}(t) \in \mathbb{R}^2$.

This dq transformation is invertible since both $\alpha\beta$ and dq reference frame do not contain information regarding the zero component. This reference frame rotates with velocity $\omega(t)$ with respect to the origin and is defined by the transformation angle $\epsilon(t)$ with $\dot{\epsilon}(t) = \omega(t)$. Also in this case, a rotating vector, which is defined by AC $\alpha\beta$ components, is transformed into a stationary vector, which is defined by DC dq components.

A.3. Transformation of Derivatives

A three-phase system is typically described as a differential equation system. Thus, time derivatives have to be transformed for obtaining a closed model in the $\alpha\beta$ or dq reference frame. In this section, the transformation of the time derivative $\dot{x}_{ph}(t) = \frac{d}{dt}x_{ph}(t)$ is shown. The $\alpha\beta0$ back-transformation is

$$\dot{x}_{ph}(t) = \frac{d}{dt}x_{ph}(t) = \frac{d}{dt}\left(\mathbf{T}_{\alpha\beta0}^{-1}x_{\alpha\beta0}(t)\right) = \mathbf{T}_{\alpha\beta0}^{-1}\frac{d}{dt}\left(x_{\alpha\beta0}(t)\right) = \mathbf{T}_{\alpha\beta0}^{-1}\dot{x}_{\alpha\beta0}(t), \tag{A.5}$$

since $\mathbf{T}_{\alpha\beta0}$ is time invariant. Thus, the full $\alpha\beta0$ transformation is

$$\begin{aligned} \dot{x}_{\alpha\beta0}(t) &= \mathbf{T}_{\alpha\beta0}\dot{x}_{ph}(t); \\ \dot{x}_{ph}(t) &= \mathbf{T}_{\alpha\beta0}^{-1}\dot{x}_{\alpha\beta0}(t); \end{aligned} \tag{A.6}$$

and the partial $\alpha\beta$ transformations is

$$\begin{aligned} \dot{x}_{\alpha\beta}(t) &= \mathbf{T}_{\alpha\beta}\dot{x}_{ph}(t); \\ \dot{x}_{ph}(t) &= \mathbf{T}_{\alpha\beta0}^{\dagger}\dot{x}_{\alpha\beta0}(t). \end{aligned} \tag{A.7}$$

On the other hand, the $dq0$ transformation is not time invariant, which yields

$$\begin{aligned} \dot{x}_{\alpha\beta0}(t) =& \frac{d}{dt}x_{\alpha\beta0}(t) = \frac{d}{dt}\left(\mathbf{T}_{dq0}^{-1}(\epsilon(t))x_{dq0}(t)\right) = \\ & \mathbf{T}_{dq0}^{-1}(\epsilon(t))\frac{d}{dt}x_{dq0}(t) + \frac{d}{dt}\left(\mathbf{T}_{dq0}^{-1}(\epsilon(t))\right)x_{dq0}(t). \end{aligned} \tag{A.8}$$

The derivation of the transformation $\mathbf{T}_{dq0}^{-1}(\epsilon(t))$ is

$$\begin{aligned}\frac{d}{dt}\mathbf{T}_{dq0}^{-1}(\epsilon(t)) =& \frac{d}{dt}\begin{bmatrix}-\sin\epsilon(t) & -\cos\epsilon(t) & 0\\ \cos\epsilon(t) & -\sin\epsilon(t) & 0\\ 0 & 0 & 0\end{bmatrix}\frac{d}{dt}\epsilon(t) =\\ & \mathbf{T}_{dq0}^{-1}(\epsilon(t))\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\omega(t) = \mathbf{T}_{dq0}^{-1}(\epsilon(t))\mathbf{J}_0\omega(t).\end{aligned} \tag{A.9}$$

Combining (A.8) and (A.9), the full $dq0$ transformation is obtained

$$\begin{aligned}\dot{x}_{dq0}(t) &= \mathbf{T}_{dq0}(\epsilon(t))\left(\dot{x}_{\alpha\beta 0}(t) - \omega(t)\mathbf{J}_0 x_{\alpha\beta 0}(t)\right);\\ \dot{x}_{\alpha\beta 0}(t) &= \mathbf{T}_{dq0}^{-1}(\epsilon(t))\left(\dot{x}_{dq0}(t) + \omega(t)\mathbf{J}_0 x_{dq0}(t)\right);\end{aligned} \tag{A.10}$$

and the partial dq transformation is

$$\begin{aligned}\dot{x}_{dq}(t) &= \mathbf{T}_{dq}(\epsilon(t))\left(\dot{x}_{\alpha\beta}(t) - \omega(t)\mathbf{J}x_{\alpha\beta}(t)\right);\\ \dot{x}_{\alpha\beta}(t) &= \mathbf{T}_{dq}^{-1}(\epsilon(t))\left(\dot{x}_{dq}(t) + \omega(t)\mathbf{J}x_{dq}(t)\right);\end{aligned} \tag{A.11}$$

where

$$\mathbf{J}_0 = \begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}; \qquad \mathbf{J} = \begin{bmatrix}0 & -1\\ 1 & 0\end{bmatrix} \tag{A.12}$$

Appendix B.

Characteristic Trajectories and Loci of the PMSM

B.1. Maximum Torque per Ampere

The maximum torque per ampere (MTPA) trajectory is derived similar to [81]. The torque equation (2.3)

$$T = \frac{3}{2} p \left(\psi + (L_d - L_q) i_d\right) i_q, \tag{B.1}$$

is rewritten using the convention

$$i_d = \|i\| \sin\gamma; \qquad i_q = \|i\| \cos\gamma; \tag{B.2}$$

and becomes

$$T = \frac{3}{2} p \left(\psi + (L_d - L_q)\|i\| \sin\gamma\right) \|i\| \cos\gamma. \tag{B.3}$$

By definition, the MTPA trajectory defines the maxima, i.e. minima, along the constant ampere ($\|i\| = \text{cnst}$) locus. Thus, the MTPA trajectory is obtained by

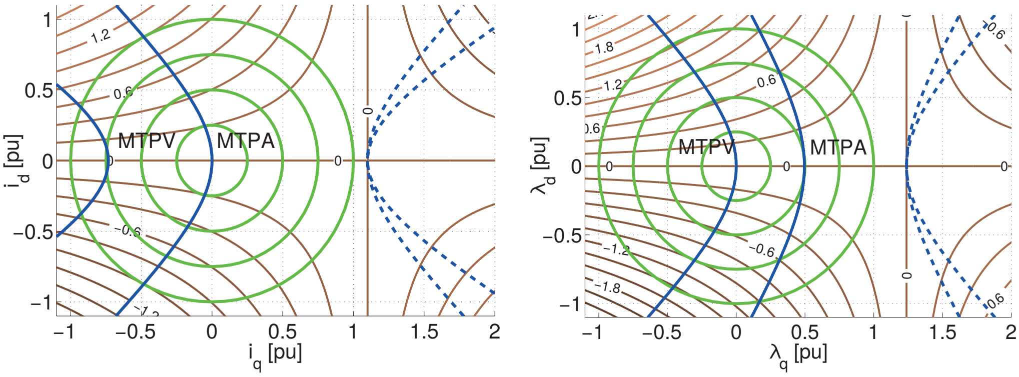
$$\frac{\partial T}{\partial \gamma} = \frac{3}{2} p \left((L_d - L_q)\|i\|^2 \cos^2\gamma - (L_d - L_q)\|i\|^2 \sin^2\gamma - \psi\|i\| \sin\gamma\right) = 0 \tag{B.4}$$

with the solution

$$\psi i_d + (L_d - L_q)\left(i_d^2 - i_q^2\right) = 0. \tag{B.5}$$

For the isotropic PMSM ($L_d - L_q = 0$), the MTPA trajectory simplifies to $i_d = 0$. Otherwise, the equation defines the hyperbola in normal form

$$\frac{\left(i_d + \frac{\psi}{2(L_d - L_q)}\right)^2}{\left(\frac{\psi}{2(L_d - L_q)}\right)^2} - \frac{i_q^2}{\left(\frac{\psi}{2(L_d - L_q)}\right)^2} = 1 \tag{B.6}$$



(a) Current space with isotorque and isocurrent

(b) Flux space with isotorque and isoflux

Figure B.1.: Global (solid) and local (dashed) MTPA and MTPV trajectory

This equation defines local maxima, i.e. minima, and it is symmetric with respect to the axis

$$i_d = -\frac{\psi}{2(L_d - L_q)}. \tag{B.7}$$

Since all anisotropic PMSM feature $L_d < L_q$ [10], it is assumed that $L_d - L_q < 0$. Thus, the symmetry axis is located on the right half plane with respect to the origin and the global maxima, i.e. minima, are located on its left side. In this condition, the MTPA trajectory is defined by

$$i_d = -\frac{\psi}{2(L_d - L_q)} - \sqrt{\left(\frac{\psi}{2(L_d - L_q)}\right)^2 + i_q^2}. \tag{B.8}$$

Sometimes, it is convenient to write the MTPA trajectory with respect to the fluxes, where it becomes

$$\lambda_d = \psi + \frac{L_d}{L_q}\left(-\frac{L_q\psi}{2(L_d - L_q)} - \sqrt{\left(\frac{L_q\psi}{2(L_d - L_q)}\right)^2 + \lambda_q^2}\right). \tag{B.9}$$

This equation shows the relation between the MTPA and MTPV trajectory and shows that the MTPA trajectory is always on the right side of the MTPV one (for $L_d < L_q$). The trajectories are shown in Figure B.1.

B.2. Maximum Torque per Volt

The maximum torque per volt (MTPV) trajectory is derived similar to [82]. The torque equation (2.4)

$$T = \frac{3}{2} p \left(\frac{\psi}{L_d} + \left(\frac{1}{L_q} - \frac{1}{L_d} \right) \lambda_d \right) \lambda_q, \tag{B.10}$$

is rewritten using the convention

$$\lambda_d = \|\lambda\| \sin\gamma; \qquad \lambda_q = \|\lambda\| \cos\gamma; \tag{B.11}$$

and becomes

$$T = \frac{3}{2} p \left(\frac{\psi}{L_d} + \left(\frac{1}{L_q} - \frac{1}{L_d} \right) \|\lambda\| \sin\gamma \right) \|\lambda\| \cos\gamma. \tag{B.12}$$

By definition, the MTPV trajectory defines the maxima, i.e. minima, along the constant volt locus. Using the compensated voltage (1.28) approach, it is equivalent to search along the constant flux ($\|\lambda\| = \text{cnst}$) locus. Thus, the MTPV trajectory is obtained by

$$\frac{\partial T}{\partial \gamma} = \frac{3}{2} p \left(\left(\frac{1}{L_q} - \frac{1}{L_d} \right) \|\lambda\|^2 \cos^2\gamma - \left(\frac{1}{L_q} - \frac{1}{L_d} \right) \|\lambda\|^2 \sin^2\gamma - \frac{\psi}{L_d} \|\lambda\| \sin\gamma \right) = 0 \tag{B.13}$$

with the solution

$$\frac{\psi}{L_d} \lambda_d + \left(\frac{1}{L_q} - \frac{1}{L_d} \right) \left(\lambda_d^2 - \lambda_q^2 \right) = 0, \tag{B.14}$$

or in analogy to the MTPA formulation

$$L_q \psi \lambda_d + (L_d - L_q) \left(\lambda_d^2 - \lambda_q^2 \right) = 0. \tag{B.15}$$

For the isotropic PMSM ($L_d - L_q = 0$), the MTPV trajectory simplifies to $\lambda_d = 0$. Otherwise, the equation defines the hyperbola in normal form

$$\frac{\left(\lambda_d + \frac{L_q \psi}{2(L_d - L_q)} \right)^2}{\left(\frac{L_q \psi}{2(L_d - L_q)} \right)^2} - \frac{\lambda_q^2}{\left(\frac{L_q \psi}{2(L_d - L_q)} \right)^2} = 1 \tag{B.16}$$

This equation defines local maxima, i.e. minima, and it is symmetric with respect to the axis

$$\lambda_d = -\frac{L_q\psi}{2(L_d - L_q)}. \tag{B.17}$$

Similar to the MTPA derivation, it is assumed that $L_d < L_q$. Thus, the symmetry axis is located on the right half plane with respect to the origin and the global maxima, i.e. minima, are located on its left side. In this condition, the MTPA trajectory is defined by

$$\lambda_d = -\frac{L_q\psi}{2(L_d - L_q)} - \sqrt{\left(\frac{L_q\psi}{2(L_d - L_q)}\right)^2 + \lambda_q^2}. \tag{B.18}$$

Sometimes, it is convenient to write the MTPA trajectory with respect to the fluxes, where it becomes

$$i_d = -\frac{\psi}{L_d} + \frac{L_q}{L_d}\left(-\frac{\psi}{2(L_d - L_q)} - \sqrt{\left(\frac{\psi}{2(L_d - L_q)}\right)^2 + i_q^2}\right). \tag{B.19}$$

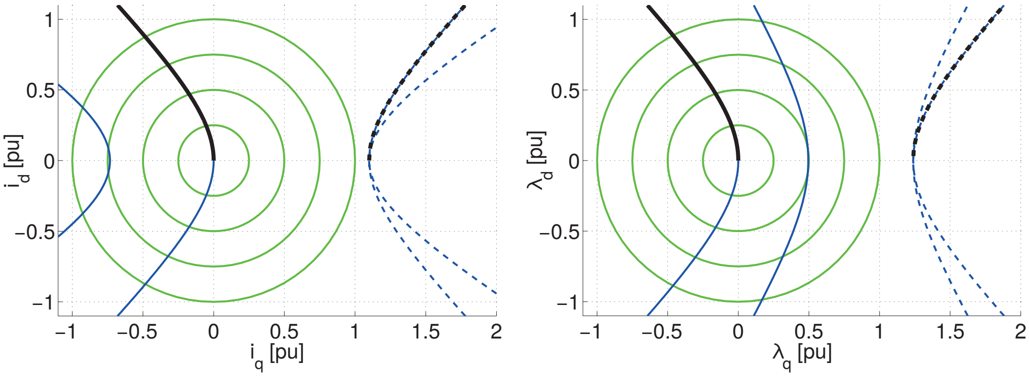
This equation shows the relation between the MTPV and MTPA trajectory and shows that the MTPV trajectory is always on the left side of the MTPA one (for $L_d < L_q$). The trajectories are shown in Figure B.1.

B.3. Intersection of MTPA and Isocurrent

It is useful to compute the intersections between trajectories. Thus, the intersections are computed in this text. If there exists an intersection, it defines two points in the current and flux space, which have the same i_d, i.e. λ_d, and the same i_q, i.e. λ_q, magnitude but opposite signs. Thus, the intersections are formulated such that an i_d is found. Moreover, the solution $i_q \geq 0$ is used when not further specified.

The MTPA and isocurrent intersection is typically interesting when computed for the largest isocurrent locus defined by the rated current I_r. This intersection defines the rated operation point $i_r = [i_{rd}, i_{rq}]^T$, where the subscript $._r$ is omitted for compactness.

For $L_d - L_q = 0$, the intersection is obtained at $i_d = 0$ and $i_q = \pm I_r$. For $L_d - L_q < 0$, the intersection is obtained substituting the isocurrent locus $i_d^2 + i_q^2 = I_r^2$ in (B.5),



(a) MTPA-isocurrent intersection varying I_r

(b) MTPV-isoflux intersection varying λ_r

Figure B.2.: Intersections for $i_q \geq 0$ and $\lambda_q \geq 0$

which yields

$$\psi i_d + (L_d - L_q)\left(2i_d^2 - I_r^2\right) = 0, \tag{B.20a}$$

$$2(L_d - L_q)i_d^2 + \psi i_d - (L_d - L_q)I_r^2 = 0. \tag{B.20b}$$

Thus it is solved by

$$i_d = -\frac{\psi}{4(L_d - L_q)} - \sqrt{\left(\frac{\psi}{4(L_d - L_q)}\right)^2 + \frac{I_r^2}{2}}, \tag{B.21}$$

$$i_q = \pm\sqrt{I_r^2 - i_d^2}. \tag{B.22}$$

The result for $i_q \geq 0$ is shown in Figure B.2(a). The result for the “false” MTPA trajectory (dashed in Figure B.2(a)) is ignored. The intersection of MTPA trajectory and isocurrent locus does always exist by definition.

B.4. Intersection of MTPV and Isoflux

The MTPV and isoflux intersection is typically interesting when computed for the largest isoflux locus defined by the reference flux $\bar{\lambda}_r \overset{\text{def}}{=} \bar{v}_r/|\omega|$. The case $|\omega| = 0$ is ignored since it has no practical relevance. In some cases (reduced power mode), this intersection defines the maximum torque defined by the states $\lambda_m = [\lambda_{md}, \lambda_{mq}]^T$, where the subscript $\cdot_m$ is omitted for compactness.

For $L_d - L_q = 0$, the intersection is obtained at $\lambda_d = 0$ and $\lambda_q = \pm\bar{\lambda}_r$. For $L_d - L_q < 0$, the intersection is obtained substituting the isocurrent locus $\lambda_d^2 + \lambda_q^2 = \bar{\lambda}_r^2$ in (B.15), which yields

$$L_q\psi\lambda_d + (L_d - L_q)\left(2\lambda_d^2 - \bar{\lambda}_r^2\right) = 0, \tag{B.23a}$$

$$2(L_d - L_q)\lambda_d^2 + L_q\psi i_d - (L_d - L_q)\bar{\lambda}_r^2 = 0. \tag{B.23b}$$

Thus it is solved by

$$\lambda_d = -\frac{L_q\psi}{4(L_d - L_q)} - \sqrt{\left(\frac{L_q\psi}{4(L_d - L_q)}\right)^2 + \frac{\bar{\lambda}_r^2}{2}}, \tag{B.24}$$

$$\lambda_q = \pm\sqrt{\bar{\lambda}_r^2 - \lambda_d^2}. \tag{B.25}$$

The result for $\lambda_q \geq 0$ is shown in Figure B.2(b). The result for the "false" MTPV trajectory (dashed in Figure B.2(b)) is ignored. The intersection of MTPV trajectory and isoflux locus does always exist by definition.

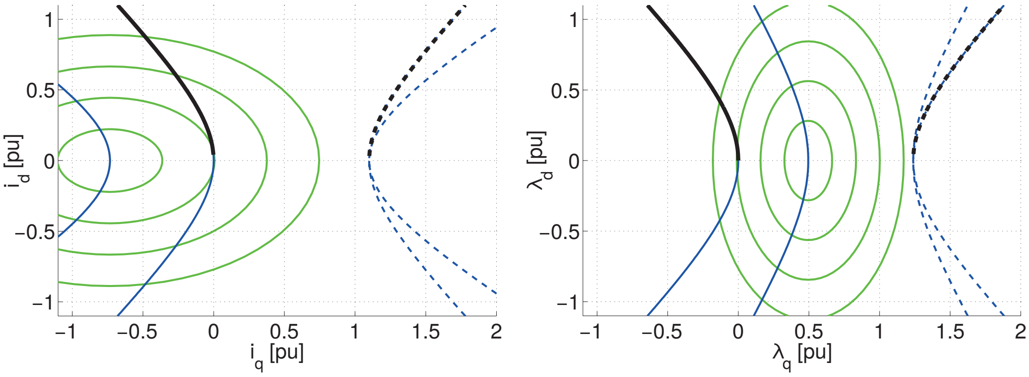
B.5. Intersection of MTPA and Isoflux

The MTPA and isoflux intersection is typically interesting when computed for the largest isoflux locus defined by the reference flux $\bar{\lambda}_r \overset{\text{def}}{=} \bar{v}_r/|\omega|$. The case $|\omega| = 0$ is ignored since it has no practical relevance. In some cases (field weakening mode), this intersection defines whether a machine should be operated on the MTPA or isoflux trajectory. This is defined by the division states $i_i = [i_{id}, i_{iq}]^T$, where the subscript $._i$ is omitted for compactness.

For $L_d - L_q = 0$, the intersection is obtained at $i_d = 0$ and $i_q = \pm 1/L_q\sqrt{\bar{\lambda}_r^2 - \psi^2}$. For $L_d - L_q < 0$, the intersection is obtained substituting the isoflux locus $\lambda_d^2 + \lambda_q^2 = \bar{\lambda}_r^2$, i.e. $(L_d i_d + \psi)^2 + (L_q i_q)^2 = \bar{\lambda}_r^2$, in (B.5), which yields

$$L_q^2\psi i_d + (L_d - L_q)\left(L_q^2 i_d^2 + (L_d i_d + \psi)^2 - \bar{\lambda}_r^2\right) = 0, \tag{B.26a}$$

$$\begin{aligned}(L_d - L_q)(L_d^2 + L_q^2)i_d^2 + \psi(2L_d^2 - 2L_dL_q + L_q^2)i_d \\ +(L_d - L_q)(\psi^2 - \bar{\lambda}_r^2) = 0.\end{aligned} \tag{B.26b}$$



(a) MTPA-isoflux intersection varying $\bar{\lambda}_r$ (b) MTPV-isocurrent intersection varying I_r

Figure B.3.: Intersections for $i_q \geq 0$ and $\lambda_q \geq 0$

Thus it is solved by

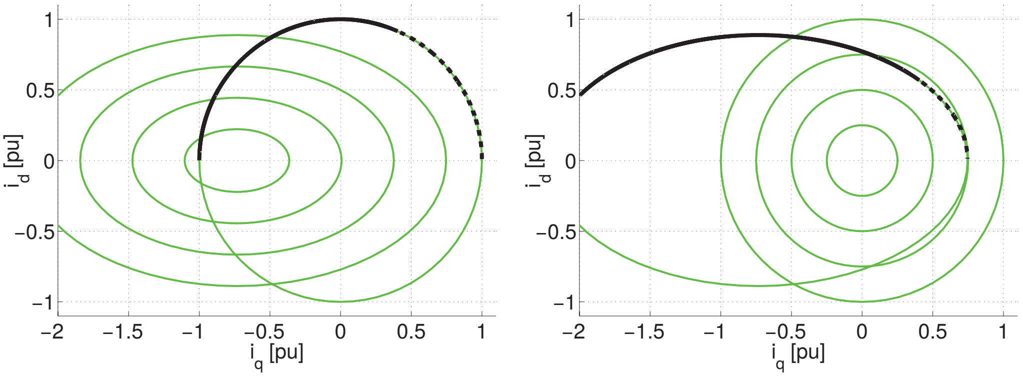
$$\begin{aligned} i_d &= -\frac{\psi(2L_d^2 - 2L_dL_q + L_q^2)}{2(L_d - L_q)(L_d^2 + L_q^2)} \\ &\quad - \sqrt{\left(\frac{\psi(2L_d^2 - 2L_dL_q + L_q^2)}{2(L_d - L_q)(L_d^2 + L_q^2)}\right)^2 + \frac{\bar{\lambda}_r^2 - \psi^2}{L_d^2 + L_q^2}}, \quad \text{(B.27)} \\ i_q &= \pm\frac{1}{L_q}\sqrt{\bar{\lambda}_r^2 - (L_d i_d + \psi)^2}. \quad \text{(B.28)} \end{aligned}$$

The result for $i_q \geq 0$ is shown in Figure B.3(a). The result for the "false" MTPA trajectory (dashed in Figure B.3(a)) is ignored. The intersection of MTPA trajectory and isoflux locus exist iff

$$\psi \leq \bar{\lambda}_r \quad \Leftrightarrow \quad |\omega|\psi \leq \bar{v}_r \tag{B.29}$$

B.6. Intersection of MTPV and Isocurrent

The MTPV and isocurrent intersection is typically interesting when computed for the largest isocurrent locus defined by the rated current I_r. This intersection defines, whether a machine works in rated power mode or reduced power mode. This is defined by the state $\lambda_p = [\lambda_{pd}, \lambda_{pq}]^T$, where the subscript $\cdot_p$ is omitted for compactness.



(a) Varying $\bar{\lambda}_r$ (b) Varying I_r

Figure B.4.: Isocurrent-isoflux intersection for $i_q \geq 0$

For $L_d - L_q = 0$, the intersection is obtained at $\lambda_d = 0$ and $\lambda_q = \pm L_q\sqrt{I_r^2 - (\psi/L_d)^2}$. For $L_d - L_q < 0$, the intersection is obtained substituting the isocurrent locus $i_d^2 + i_q^2 = I_r^2$, i.e. $L_q^2(\lambda_d - \psi)^2 + L_d^2\lambda_q^2 = L_d^2 L_q^2 I_r^2$, in (B.15), which yields

$$L_d^2 L_q \psi \lambda_d + (L_d - L_q)\left(L_d^2\lambda_d^2 + L_q^2(\lambda_d - \psi)^2 - L_d^2 L_q^2 I_r^2\right) = 0, \tag{B.30a}$$

$$\begin{aligned}(L_d - L_q)(L_d^2 + L_q^2)\lambda_d^2 + L_q\psi(L_d^2 - 2L_dL_q + 2L_q^2)\lambda_d \\ + L_q^2(L_d - L_q)(\psi^2 - L_d^2 I_r^2) = 0.\end{aligned} \tag{B.30b}$$

Thus it is solved by

$$\begin{aligned}\lambda_d = &-\frac{L_q\psi(L_d^2 - 2L_dL_q + 2L_q^2)}{2(L_d - L_q)(L_d^2 + L_q^2)} \\ &- \sqrt{\left(\frac{L_q\psi(L_d^2 - 2L_dL_q + 2L_q^2)}{2(L_d - L_q)(L_d^2 + L_q^2)}\right)^2 + \frac{L_q^2(L_d^2 I_r^2 - \psi^2)}{L_d^2 + L_q^2}},\end{aligned} \tag{B.31}$$

$$\lambda_q = \pm L_q\sqrt{I_r^2 - \left(\frac{\lambda_d}{L_d} - \frac{\psi}{L_d}\right)^2}. \tag{B.32}$$

The result for $i_q \geq 0$ is shown in Figure B.3(b). The result for the "false" MTPV trajectory (dashed in Figure B.3(b)) is ignored. The intersection of MTPV trajectory and isocurrent locus exist iff

$$\frac{\psi}{L_d} \leq I_r \tag{B.33}$$

B.7. Intersection of Isocurrent and Isoflux

The isocurrent and isoflux intersection is typically interesting when computed for the largest isocurrent locus defined by I_r and the largest isoflux locus defined by the reference flux $\bar{\lambda}_r \overset{\text{def}}{=} \bar{v}_r/|\omega|$. The case $|\omega| = 0$ is ignored since it has little practical relevance. In some cases (constant power mode), this intersection defines the maximum torque defined by the states $\lambda_m = [\lambda_{md}, \lambda_{mq}]^T$, where the subscript $\cdot_m$ is omitted for compactness.

The intersection is obtained substituting the isocurrent locus $i_d^2 + i_q^2 = I_r^2$ in the isoflux locus $(L_d i_d + \psi)^2 + (L_q i_q)^2 = \bar{\lambda}_r^2$, which yields

$$(L_d^2 - L_q^2)i_d^2 + 2L_d\psi i_d + \psi^2 - \bar{\lambda}_r^2 + L_q^2 I_r^2 = 0. \tag{B.34}$$

For $L_d - L_q = 0$, this equation defines the intersection at

$$i_d = \frac{\bar{\lambda}_r^2 - L_q^2 I_r^2 - \psi^2}{2L_q\psi}, \tag{B.35}$$

$$i_q = \pm\sqrt{I_r^2 - i_d^2}. \tag{B.36}$$

For $L_d - L_q < 0$, the intersection is at

$$i_d = \frac{-L_d\psi + \sqrt{L_q^2\psi^2 + (L_d^2 - L_q^2)(\bar{\lambda}_r^2 - L_q^2 I_r^2)}}{L_d^2 - L_q^2}, \tag{B.37}$$

$$i_q = \pm\sqrt{I_r^2 - i_d^2}. \tag{B.38}$$

The result for $i_q \geq 0$ is shown in Figure B.4. The second result at $i_d > 0$ (dashed in Figure B.4) is ignored since a PMSM is not operated in this region. The existence of the isocurrent and isoflux intersection depends on several cases. To the aim of this text it is sufficient observing that the intersection always exists when the machine is operated in constant power mode.

B.8. Computing a Torque on the MTPA Trajectory

When controlling a drive system it is necessary to compute current, i.e. flux, references from a torque reference. Clearly, there does not exist a unique current i_{dq} which produces the requested torque T. However, the optimal state $i_{dq}^\star$, which produces the requested torque with the smallest losses, is defined. The $i_{dq}^\star$ lies on the MTPA trajectory or the largest isoflux locus br Λ dependent on the operation conditions and the magnitude of the torque. In this section, the $i^\star_{dq}$, which produced defined torque T and lies on the MTPA trajectory, is computed. Similar to the trajectory and locus intersections, the $i^\star_d$ is computed first, which identifies the operation point. Then, the corresponding $i^\star_q$ is computed.

For isotropic machines ($L_d - L_q = 0$), the q axis current is defined by the torque equation (2.3) is $T = 3/2p\psi i_q$ and the MTPA trajectory (B.5) simplifies to $i_d = 0$ and. Thus, the optimal operation point on the MTPA trajectory is defined by

$$i^\star_q = \frac{T}{3/2p\psi}, \tag{B.39a}$$

$$i^\star_d = 0. \tag{B.39b}$$

For anisotropic machines ($L_\Delta \overset{\text{def}}{=} L_d - L_q < 0$), the transformed MTPA trajectory (B.5) $L_\Delta i_q^2 = i_d(\psi + L_\Delta i_d)$ is substituted in the transformed torque equation (2.3) $(3/2p)^2 \left(\psi + L_\Delta i_d\right)^2 L_\Delta i_q^2 - L_\Delta T^2 = 0$, which yields

$$(3/2p)^2 \left(\psi + L_\Delta i_d\right)^3 i_d - L_\Delta T^2 = 0, \tag{B.40a}$$

$$(3/2p)^2 \left(L_\Delta^3 i_d^4 + 3L_\Delta^2 \psi i_d^3 + 3L_\Delta \psi^2 i_d^2 + \psi^3 i_d\right) - L_\Delta T^2 = 0. \tag{B.40b}$$

This quartic equation is solved analytically using e.g. Ferrari's method. A quartic equation has up to four roots, which can be either real or complex. However, by definition of the problem there exists only one real real root, which satisfies $i_d \leq 0$. This solution corresponds to $i^\star_d$. The other current $i^\star_q$ is found by substitution in (2.3).

B.9. Computing a Torque on the Isoflux Locus

In this section, the $i^\star_{dq}$, which produced defined torque T and lies on the isoflux locus, is computed. Before starting, it is observed that not any torque $T \in \mathbb{R}$ can be produced by states on a isoflux trajectory but only T with a limited magnitude. The maximum (and minimum) torque T_m (and $-T_m$) is obtained at the intersection of the MTPV trajectory and the isoflux trajectory. Thus, it is known a priori if T can be obtained and $T \leq T_m$ is assumed when solving the problem. This implies the existence of a real solution. Moreover, operation on the isoflux trajectory is only interesting when operating in field weakening mode, where $|\omega| > 0$. In this case, operation on the maximum flux magnitude $\lambda_r = \bar{v}_r/|\omega|$ is desired.

For isotropic machines ($L_d - L_q = 0$), the q axis current is defined by the torque equation (2.3) is $T = 3/2p\psi i_q$. Intersecting this i_q current with the isoflux locus, two roots are obtained, that are located one on the left and the right side of the MTPV trajectory. Operation on the right side is desired due to the lower current magnitude, which yields the solution

$$i_q^\star = \frac{T}{3/2p\psi}, \tag{B.41a}$$

$$i_d^\star = -\frac{\psi}{L_d} + \frac{1}{L_d}\sqrt{\lambda_r^2 - L_d^2 i_q^2}. \tag{B.41b}$$

For anisotropic machines ($L_\Delta \overset{\text{def}}{=} L_d - L_q < 0$), the transformed isoflux locus $L_q^2 i_q^2 = \lambda_r - (L_d i_d + \psi)^2$ is substituted in the transformed torque equation (2.3) $(3/2p)^2 \left(\psi + L_\Delta i_d\right)^2 L_q^2 i_q^2 - L_q^2 T^2 = 0$, which yields

$$(3/2p)^2 \left(\psi + L_\Delta i_d\right)^2 \left(\lambda_r - (L_d i_d + \psi)^2\right) - L_q^2 T^2 = 0, \tag{B.42a}$$

$$\begin{aligned}(3/2p)^2(-L_d^2 L_\Delta^2 i_d^4 - 2L_d\psi L_\Delta(2L_d - Lq)i_d^3 + \\ (\lambda_r^2 L_\Delta^2 - \psi^2(6L_d^2 - 6L_d L_q + L_q))i_d^2 + \\ 2\psi(\lambda_r^2 L_\Delta - \psi^2(2L_d - L_q))i_d + \psi^2(\lambda_r^2 - \psi^2)) - L_q^2 T^2 = 0.\end{aligned} \tag{B.42b}$$

This quartic equation is solved analytically using e.g. Ferrari's method and has up to four roots, which can be either real or complex. In conditions where operation on the isoflux trajectory is interesting, there are (at least) two real roots. The optimal result $i_d^\star$ is real and located on the right side of the MTPV trajectory and on the left side of the MTPA trajectory. By definition of the problem, there is only one root satisfying this conditions and $i_q^\star$ is found by substitution in (2.3).

Appendix C.

Observer

Any control algorithm has a finite execution time. In power electronic systems, the execution time of the control code is typically not small compared to the sampling period. Thus, a unit delay is introduced between measurement and actuation. Consequently, the state $x[k]$ is not available for control and the plant input $u[k]$ has to be computed based on the available state $x[k-1]$. This delay should be compensated since it is critical for the control performance. In other words, the control algorithm should not consider control errors, which have been taken care of but have not been “seen” by the measurement so far.

C.1. System

In this chapter, a state-space formulation based on the discrete-time model (1.20) is used in the dq reference frame. Let the system dynamics be

$$x[k] = \mathbf{A}x[k-1] + \mathbf{B}u[k-1] + w[k-1], \tag{C.1}$$

where the matrices $\mathbf{A}$ and $\mathbf{B}$ are obtained via exact (or zero-order-hold) discretization (1.21), Tustin discretization (1.23), or also Euler discretization (1.24). The system state is the dq reference frame flux $x = \lambda_{dq}$ and the system input is the (already applied) dq reference frame terminal voltage $u = v_{dq}$. The disturbance w takes the non-modeled system behavior into account: the resistive voltage drop, inverter nonlinear effects, measurement noise, high-frequency dynamics due to parasitic capacitances, etc.

This model is used to design an observer with the goal to reduce the effect of the unit delay and not modeled behavior. The observer can be designed and computed independently from control due to the separation principle [40], [101]. Several observer designs and implementations have been proposed in literature. Most popular are the Luenberger state observer [60] and the Kalman observer [48].

In this chapter, the former one is used, due to its simplicity and performance. The Luenberger observer has been applied successfully to drive systems in [80]–[82]. However, it is observed that the Kalman observer can improve performance if the measurement (white) noise level is critical. This improvement is achieved at the price of an increased design and implementation complexity.

C.2. Design

In this section, the Luenberger state observer is designed and an intuitive tuning method is proposed. For the system (C.1), the observer is written as

$$\hat{x}[k] = \mathbf{A}\hat{x}[k-1] + \mathbf{B}u[k-1] - \mathbf{G}\hat{\hat{x}}[k-1], \tag{C.2}$$

where $\hat{x}$ is the estimate, $\hat{\hat{x}} = \hat{x} - x$ is the estimation error, and $\mathbf{G}$ is the Luenberger gain.

Theorem C.1. *Choosing $\mathbf{G} \overset{def}{=} \mathbf{A} - (1-G)\mathbf{I}$, the observer (C.2) is asymptotically stable, i.e. $\hat{\hat{x}} \to 0$ for $k \to \infty$, iff*

$$G \in (0, 2). \tag{C.3}$$

Proof. Subtracting the dynamics (C.1) from the observer equation (C.2), the observation error dynamic is obtained

$$\hat{\hat{x}}[k] = (\mathbf{A} - \mathbf{G})\hat{\hat{x}}[k-1] - w[k-1], \tag{C.4}$$

which is stable if the eigenvalues of $\mathbf{A} - \mathbf{G}$ are in the unit circle. Choosing $\mathbf{G} = \mathbf{A} - (1-G)\mathbf{I}$ yields

$$\hat{\hat{x}}[k] = (1-G)\hat{\hat{x}}[k-1] - w[k-1]. \tag{C.5}$$

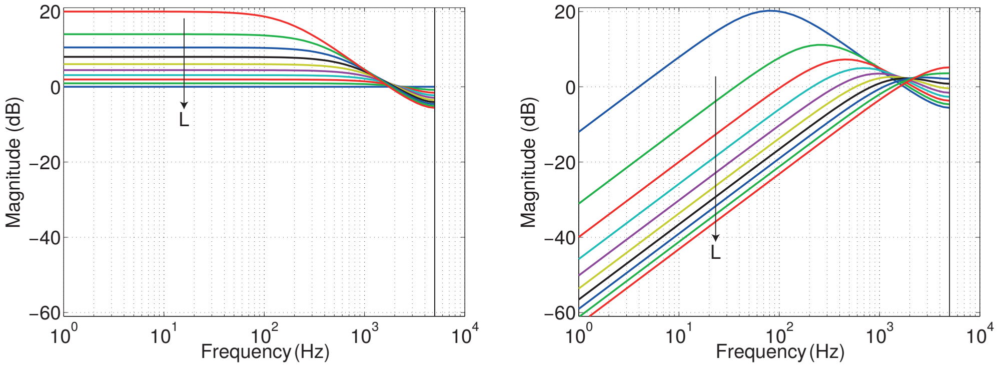
with the eigenvalue(s) $1 - G$. Thus, the system is asymptotically stable iff

$$1 - G \in (-1, 1) \quad \Leftrightarrow \quad G \in (0, 2) \tag{C.6}$$

□

Using the observer (C.2), the disturbance to estimation error transfer function is

$$\frac{\hat{\hat{x}}_d}{w_d} = \frac{\hat{\hat{x}}_q}{w_q} = \frac{1}{(1-G) - z}, \tag{C.7}$$



(a) Without integral gain

(b) With integral gain

Figure C.1.: Disturbance to state estimate transfer function $\hat{\tilde{x}}_d/w_d = \hat{\tilde{x}}_q/w_q$ for $T_s = 10^{-4}$ (note: the transfer function moves with the sampling frequency maintaining its shape in log-scale)

which is shown in Figure C.1(a). Slow varying disturbances and offsets are not only passed to the estimate but are amplified. Thus, $\hat{\tilde{x}}$ does not converge to zero if w is nonzero. Since an estimation bias results in a control offset for any controller, it has to be removed. Offsets can be removed augmenting the observer function with the error integral, which yields

$$\hat{x}[k] = \mathbf{A}\hat{x}[k-1] + \mathbf{B}u[k-1] - \mathbf{G}\hat{\tilde{x}}[k-1] - \mathbf{G}_i\hat{\tilde{x}}_i[k-1], \tag{C.8a}$$

$$\hat{\tilde{x}}_i[k] = \hat{\tilde{x}}_i[k-1] + T_s\hat{\tilde{x}}[k-1] \tag{C.8b}$$

Regarding stability of the modified observer, the following result is obtained.

Theorem C.2. *Choosing* $\mathbf{G} \overset{def}{=} \mathbf{A} - (1-G)\mathbf{I}$ *and* $\mathbf{G}_i \overset{def}{=} G_i\mathbf{I}$*, the observer (C.8) is asymptotically stable, i.e.* $\hat{\tilde{x}} \to 0$ *for* $k \to \infty$*, if*

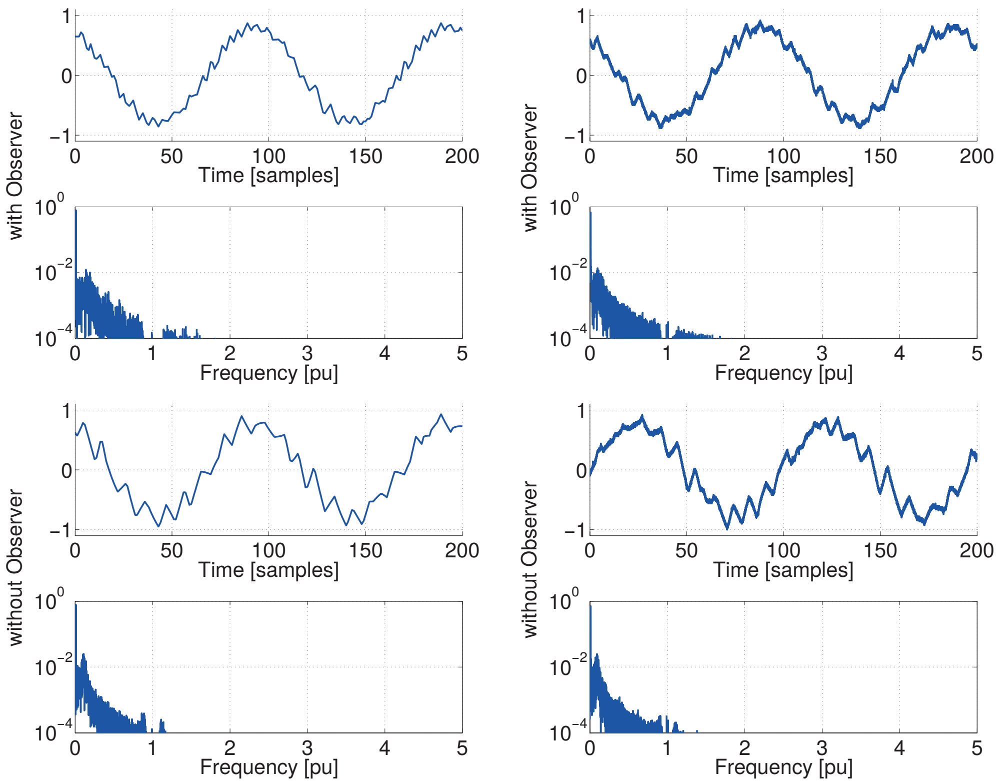
$$G_i = \frac{G^2}{4T_s} \quad \Leftrightarrow \quad G \in (0,4). \tag{C.9}$$

Proof. Subtracting the dynamics (C.1) from the observer equation (C.8), the observation error dynamic is obtained

$$\begin{bmatrix} \hat{\tilde{x}}[k] \\ \hat{\tilde{x}}_i[k] \end{bmatrix} = \begin{bmatrix} \mathbf{A}-\mathbf{G} & -\mathbf{G}_i \\ T_s\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \hat{\tilde{x}}[k-1] \\ \hat{\tilde{x}}_i[k-1] \end{bmatrix} - \begin{bmatrix} w[k-1] \\ \end{bmatrix} \tag{C.10}$$

Choosing $\mathbf{G} = \mathbf{A} - (1-G)\mathbf{I}$ and $\mathbf{G}_i = G_i\mathbf{I}$ yields

$$\begin{bmatrix} \hat{\tilde{x}}[k] \\ \hat{\tilde{x}}_i[k] \end{bmatrix} = \begin{bmatrix} (1-G)\mathbf{I} & -G_i\mathbf{I} \\ T_s\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \hat{\tilde{x}}[k-1] \\ \hat{\tilde{x}}_i[k-1] \end{bmatrix} - \begin{bmatrix} w[k-1] \\ \end{bmatrix} \tag{C.11}$$



(a) Simulation

(b) Experimentation

Figure C.2.: FCS-MPC current quality with and without observer

with the eigenvalues

$$\operatorname{eig}\begin{bmatrix}(1-G)\mathbf{I} & -G_i\mathbf{I}\\ T_s\mathbf{I} & \mathbf{I}\end{bmatrix} = 1-\frac{G}{2}\pm\sqrt{\left(\frac{G}{2}\right)^2 - G_iT_s}. \tag{C.12}$$

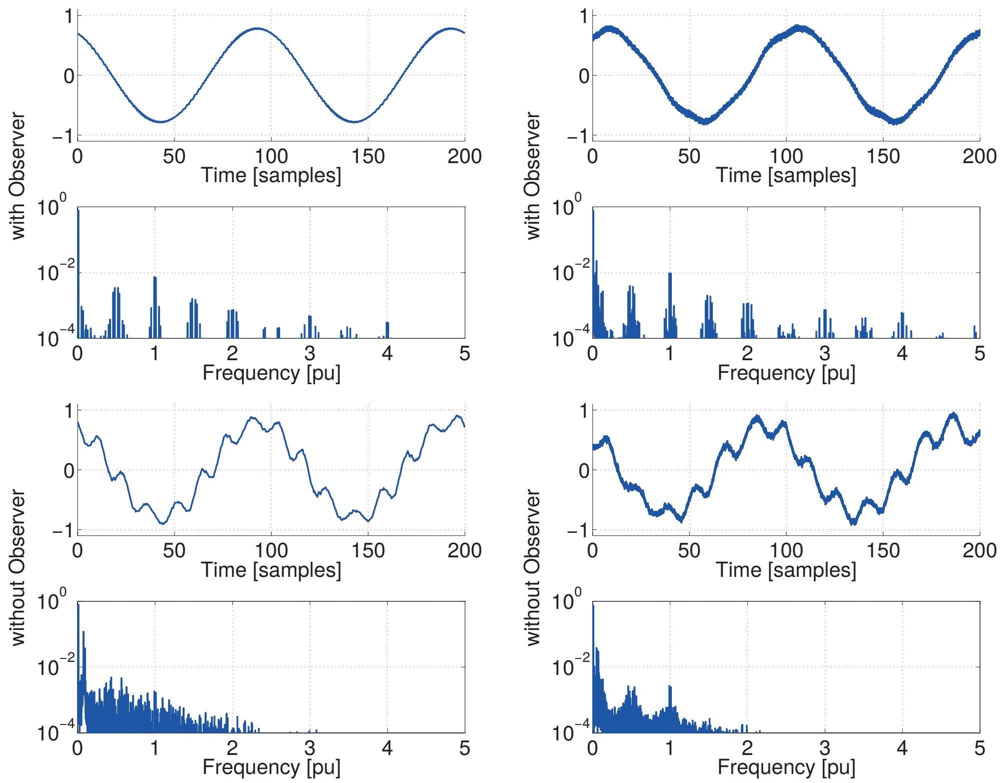
The observer is asymptotically stable iff $\Re\{\text{eig}[.]\}^2 + \Im\{\text{eig}[.]\}^2 < 1$. Among other solutions, this goal is achieved if $G_i = \frac{G^2}{4T_s}$, such that the eigenvalue(s) become

$$1-\frac{G}{2}\pm\sqrt{\left(\frac{G}{2}\right)^2 - \frac{G^2}{4T_s}T_s} = 1-\frac{G}{2}. \tag{C.13}$$

The resulting system is asymptotically stable under the condition

$$1-\frac{G}{2}\in(-1,1) \quad\Leftrightarrow\quad G\in(0,4). \tag{C.14}$$

□



(a) Simulation

(b) Experimentation

Figure C.3.: CCS-MPC current quality with and without observer

The observer (C.8) with $G_i = \frac{G^2}{4T_s}$ yields the disturbance to estimation error transfer function

$$\frac{\hat{\tilde{x}}_d}{w_d} = \frac{\hat{\tilde{x}}_q}{w_q} = \frac{1-z}{\left(\left(1-\frac{G}{2}\right)-z\right)^2}, \tag{C.15}$$

which is shown in Figure C.1(b). In this case, a disturbance bias has no effect on the estimation error and slow varying disturbances have few influence. However, disturbances with higher frequencies can be amplified dependent on the choice of the observer gain G. In fact, the choice of G can be interpreted as follows. For low values of G, the observer does not reject well disturbances of the mid-frequency range but is less sensitive to noise close to the Nyquist frequency. Increasing G, mid-frequency disturbances are increasingly better rejected at the price of a higher noise sensitivity.

C.3. Evaluation

The observer has been used to achieve the results in Chapter 4, Chapter 5, and Chapter 6 on both, the experimental and simulation platform outlined in Appendix D. Consequently, the observer itself is not evaluated but deterioration effects are shown if the observer is omitted.

Such effects are best visible on FCS-MPC, where the control error is small, i.e. in steady-state operation. If the observer is omitted, FCS-MPC attempts to compensate an old control error, which disregards the last control action. Since this control action steers the error to a different location, which the controller is not aware of, the current quality deteriorates significantly. This effect, is shown in Figure C.2.

On CCS-MPC (or nonlinear control), the unit delay is observed to have a different effect. If the sampling frequency is sufficiently high the observer may be omitted without significant deterioration effects on the system. If the sampling frequency is low, the unit delay can introduce oscillations, which are shown in Figure C.3.

Appendix D.

Simulation and Experimentation Platform

In this text, the presented concepts are evaluated using a simulation platform and an experimental test bench. The control algorithms and procedures are implemented in ANSI-C (C99) and the same code is executed on both platforms. The main parameters of the setup are shown in Table D.1.

Table D.1.: Drive System Parameters

Inverter and Control	
Type	two level voltage source inverter (2L-VSI)
Grid interface	diode bridge
Interlock time T_i	$3\mu s$
DC-link voltage v_c	$120V$
Embedded control platform	dSpace 1104
Sampling time T_s	$200\mu s$
Electric Machine	
Type	interior PM synchronous machine (IPMSM)
Rated current I_s	$10A$
Rated torque T_r	$8.0Nm$
Rated flux λ_r	$142.5mWb$
Inductance (d-axis) L_d	$9.1mH$
Inductance (q-axis) L_q	$14.6mH$
Stator resistance R_s	$636m\Omega$
PM rotor flux ψ	$88.3mWb$
Pole pairs p (model, physical)	5.3, 5
Shaft friction B	$6.4 \cdot 10^{-3}Nms$
Shaft inertia J	$5.0 \cdot 10^{-3}kgm^2$

D.1. Simulation Platform

The control concepts and algorithms are developed and implemented using a simulation platform. The platform is implemented using the Software-in-the-Loop (SiL) concept. It uses Matlab and Simulink calling control code via *mex* and *s-function* interfaces. In Simulink, a high fidelity drive system model is implemented to represent the plant. Besides the typical equations of inverter and electric machine, the model includes also secondary effects. The PMSM model is implemented using the measured nonlinear flux-current relationship. Mechanically, an approximately stiff drive shaft is considered, which is characterized by a concentrated inertia, friction coefficient, and friction torque.

The inverter is represented by its switch model adding the dominant nonlinear effects, i.e. the voltage drops and interlock times. The control code is executed at each sampling instant with sampling period T_s. The hardware models are sampled at least two orders of magnitude faster than T_s to capture properly the system behavior, e.g. the PWM. The SiL platform has been used and validated experimentally during several projects, e.g. [80]–[82], where the SiL results have shown good correspondence with the experimental data.

The PC platform running the SiL model is also used to make comparisons in terms of computational complexity and execution time of control code. This platform consists of an INTEL i7 620M CPU, 8GB RAM, and Win7 OS, where the timed task is executed with *real time* priority. Clearly, evaluating execution times on a modern PC platform is subject to uncertainties. To minimize these effects, the algorithms are executed repeatedly and minimum, mean, and maximum timings are presented. This evaluation is not carried out to provide exact timings but to identify viable approaches for the experimental implementation.

D.2. Experimental Test Bench

The experimental test bench is used to show that the proposed concepts can be implemented and executed in real time. Moreover, it is used to confirm the obtained results experimentally in the presence of non-modeled system behavior, e.g. high frequency capacitances. The test bench consists of two back-to-back drive systems. One is a standard industrial drive, which is used to generate a load torque T_l. On the other drive system, the proposed control developments are tested. The developed control code is compiled for the embedded control hardware, which executes the code in real time.

The control code is executed on a dSpace 1104, which is used as embedded control hardware. This system consists of a $300MHz$ PowerPC *master* chip, which is used to elaborate the real time data, and a $20MHz$ TI TMS320F240 *slave* chip, which provides advanced interfaces, e.g. DAC and PWM. The dSpace is used due to its fast prototyping capabilities, i.e. the ability to display, store, and manipulate a high amount of data at sampling frequency. However, it has also some device specific drawbacks. The implementation of the master-slave communication requires that the system input, i.e. the duty cycles of the PWM module, are set before half the sampling period has passed or the input is actuated with an additional sampling period of delay. Moreover, the system is halted during measurement acquisition by the slave module and communication to the master. As a consequence, only a part of the computation capabilities can be used to execute control code, which yields a mediocre computation performance compared to typical industrial DSP, e.g. the TI C28x Piccolo or Delphino series.

Bibliography

[1] P. M. Anderson and A. A. Fouad, *Power System Control and Stability.* Wiley-IEEE Press, 2003, ISBN: 9780471238621.

[2] M. Andriollo, G. Martinelli, and A. Morini, *Macchine Elettriche Rotanti.* Cortina (Padua), 2009, ISBN: 9788877843074.

[3] Z. Artstein, «Stabilization with relaxed controls», *Nonlinear Analysis: Theory, Methods & Applications*, vol. 7, pp. 1163–1173, 1983. DOI: `10.1016/0362-546X(83)90049-4`.

[4] L. Asiminoaei, P. Rodriguez, and F. Blaabjerg, «Application of discontinuous pwm modulation in active power filters», *IEEE Transactions on Power Electronics*, vol. 23, pp. 1692–1706, 2008. DOI: `10.1109/TPEL.2008.924599`.

[5] M. Barcaro, «Design and analysis of interior permanent magnet synchronous machines for electric vehicles», PhD thesis, University of Padua, Italy, 2011. [Online]. Available: `http://paduaresearch.cab.unipd.it/3497/`.

[6] M. Barcaro, E. Fornasiero, N. Bianchi, and S. Bolognani, «Design procedure of ipm motor drive for railway traction», in *International Electric Machines and Drives Conference (IEMDC)*, 2011. DOI: `10.1109/IEMDC.2011.5994950`.

[7] M. M. Bech, J. K. Pedersen, and F. Blaabjerg, «Random modulation techniques with fixed switching frequency for three-phase power converters», *IEEE Transactions on Power Electronics*, vol. 15, pp. 753–761, 2000. DOI: `10.1109/63.849046`.

[8] M. M. Bech, J. K. Pedersen, F. Blaabjerg, and A. M. Trzynadlowski, «A methodology for true comparison of analytical and measured frequency domain spectra in random pwm converters», *IEEE Transactions on Power Electronics*, vol. 14, pp. 578–586, 1999. DOI: `10.1109/63.761702`.

[9] A. Benchaib, S. Poullain, J.-L. Thomas, and J. Alacoque, «Discrete-time field-oriented control for sm-pmsm including voltage and current constraints», in *International Electric Machines and Drives Conference (IEMDC)*, 2003. DOI: `10.1109/IEMDC.2003.1210357`.

[10] N. Bianchi and S. Bolognani, «Parameters and volt-ampere ratings of a synchronous motor drive for flux-weakening applications», *IEEE Transactions on Power Electronics*, vol. 12, pp. 895–903, 1997. DOI: `10.1109/63.623008`.

[11] ——, *Metodologie di Progettazione delle Macchine Elettriche*. Cleup (Padua), 2001, ISBN: 9788871785295.

[12] N. Bianchi, S. Bolognani, and B. Ruzojcic, «Design of a 1000 hp permanent magnet synchronous motor for ship propulsion», in *European Conference Power Electronics and Applications (EPE)*, 2009.

[13] N. Bianchi and T. M. Jahns, *Design, Analysis, and Control of Interior PM Synchronous Machines*. Cleup (Padua), 2006, ISBN: 9788871788982.

[14] J. L. Blackburn, *Symmetrical Components for Power Systems Engineering*. Marcel Dekker, New York, 1993, ISBN: 9780824787677.

[15] F. Blanchini, «Ultimate boundedness control for uncertain discrete-time systems via set-induced lyapunov functions», *IEEE Transactions on Automatic Control*, vol. 39, pp. 428–433, 1994. DOI: 10.1109/9.272351.

[16] ——, «Set invariance in control», *Automatica*, vol. 35, pp. 1747–1767, 1999. DOI: 10.1016/S0005-1098(99)00113-2.

[17] I. Boldea and S. Nasar, «A general equivalent circuit (gec) of electric machines including crosscoupling saturation and frequency effects», *IEEE Transactions on Energy Conversion*, vol. 3, pp. 689–695, 1988. DOI: 10.1109/60.8086.

[18] S. Bolognani, S. Calligaro, R. Petrella, and F. Pogni, «Flux-weakening in ipm motor drives: comparison of state-of-art algorithms and a novel proposal for controller design», in *European Conference Power Electronics and Applications (EPE)*, 2011.

[19] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2011. [Online]. Available: `http://www.mpc.berkeley.edu/mpc-course-material`.

[20] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004, ISBN: 9780521833783. [Online]. Available: `http://www.stanford.edu/~boyd/cvxbook/`.

[21] M. Carraro, F. Tinazzi, and M. Zigliotto, «A novel approach to torque estimation in ipm synchronous motor drives», in *Annual Conference of the IEEE Industrial Electronics Society (IECON)*, 2012. DOI: 10.1109/IECON.2012.6389499.

[22] ——, «Estimation of the direct-axis inductance in pm synchronous motor drives at standstill», in *International Conference on Industrial Technology (ICIT)*, 2013. DOI: 10.1109/ICIT.2013.6505691.

[23] G. Casadei, «Modular multi-level converter: modeling, simulation and control in steady state and dynamic conditions», Master's thesis, Aalborg University, Denmark, 2012. [Online]. Available: `http://projekter.aau.dk/projekter/files/63645060/THESISrevFinale.pdf`.

[24] C.-T. Chen, *Linear System Theory and Design*. Oxford University Press, 1999, ISBN: 9780199959570.

[25] C.-H. Choi and J.-K. Seok, «Compensation of zero-current clamping effects in high-frequency-signal-injection-based sensorless pm motor drives», *IEEE Transactions on Industry Applications*, vol. 43, pp. 1258–1265, 2007. DOI: `10.1109/TIA.2007.904409`.

[26] P. D. Christofides, J. Liu, and D. Muñoz de la Peña, *Lyapunov-Based Model Predictive Control*. Springer, 2011, ISBN: 9780857295811. DOI: `10.1007/978-0-85729-582-8_2`.

[27] P. Cortés, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodríguez, «Predictive control in power electronics and drives», *IEEE Transactions on Industrial Electronics*, vol. 55, pp. 4312–4324, 2008. DOI: `10.1109/TIE.2008.2007480`.

[28] R. DeCarlo, *Linear Systems: A State Variable Approach with Numerical Implementation*. Prentice Hall, 1989, ISBN: 9780135368145.

[29] M. Degner and R. Lorenz, «Using multiple saliencies for the estimation of flux, position, and velocity in ac machines», *IEEE Transactions on Industry Applications*, vol. 34, pp. 1097–1104, 1998. DOI: `10.1109/28.720450`.

[30] A. Domahidi, A. Zgraggen, M. Zeilinger, M. Morari, and C. Jones, «Efficient interior point methods for multistage problems arising in receding horizon control», in *Conference on Decision and Control (CDC)*, 2012. DOI: `10.1109/CDC.2012.6426855`. [Online]. Available: `http://forces.ethz.ch/`.

[31] W. Duesterhoeft, M. Schulz, and E. Clarke, «Determination of instantaneous currents and voltages by means of alpha, beta, and zero components», *Transactions of the American Institute of Electrical Engineers*, vol. 70, pp. 1248–1255, 1951. DOI: `10.1109/T-AIEE.1951.5060554`.

[32] R. W. Erickson and D. Maksimovic, *Fundamentals of Power Electronics*. Springer, 2011, ISBN: 9781475705591.

[33] R. Fischer, *Elektrische Maschinen*. Hanser, 2011, ISBN: 9783446425545.

[34] R. Gabriel, W. Leonhard, and C. J. Nordby, «Field-oriented control of a standard ac motor using microprocessors», *IEEE Transactions on Industry Applications*, vol. IA-16, pp. 186–192, 1980. DOI: `10.1109/TIA.1980.4503770`.

[35] E. M. Gertz and S. J. Wright, «Object-oriented software for quadratic programming», *ACM Transactions on Mathematical Software*, vol. 29, pp. 58–81, 2003. DOI: 10.1145/641876.641880. [Online]. Available: http://pages.cs.wisc.edu/~swright/ooqp/.

[36] T. Geyer, «Computationally efficient model predictive direct torque control», *IEEE Transactions on Power Electronics*, vol. 26, pp. 2804–2816, 2011. DOI: 10.1109/TPEL.2011.2121921.

[37] J. Guerrero, M. Leetmaa, F. Briz, A. Zamarron, and R. Lorenz, «Inverter nonlinearity effects in high-frequency signal-injection-based sensorless control methods», *IEEE Transactions on Industry Applications*, vol. 41, pp. 618–626, 2005. DOI: 10.1109/TIA.2005.844411.

[38] R. F. Harrington, *Introduction to Electromagnetic Engineering*. Dover Publications (Mineola), 2003, ISBN: 9780486432410.

[39] F. Haugen, «Discrete-time signals and systems», TechTeach, Norway, Tech. Rep., 2005. [Online]. Available: http://techteach.no/publications/discretetime_signals_systems/discrete.pdf.

[40] J. P. Hespanha, «Lqg/lqr controller design», University of California Santa Barbara, USA, Tech. Rep., 2007. [Online]. Available: www.ece.ucsb.edu/~hespanha/ece147c-me106a/web/lqrlqgnotes.pdf.

[41] J. Holtz, «Pulsewidth modulation-a survey», *IEEE Transactions on Industrial Electronics*, vol. 39, pp. 410–420, 1992. DOI: 10.1109/41.161472.

[42] J. Holtz and J. Quan, «Sensorless vector control of induction motors at very low speed using a nonlinear inverter model and parameter identification», *IEEE Transactions on Industry Applications*, vol. 38, pp. 1087–1095, 2002. DOI: 10.1109/TIA.2002.800779.

[43] J. Holtz and S. Stadtfeld, «A predictive controller for the stator current vector of ac machines fed from a switched voltage source», in *International Power Electronic Conference (IPEC)*, 1983.

[44] J. Huang, K. Padmanabhan, and O. M. Collins, «The sampling theorem with constant amplitude variable width pulses», *IEEE Transactions on Circuits and Systems*, vol. 58, pp. 1178–1190, 2011. DOI: 10.1109/TCSI.2010.2094350.

[45] IBM. (2013). ILOG CPLEX optimization studio, [Online]. Available: http://ibm.com/software/commerce/optimization/cplex-optimizer/.

[46] K. Ide, S. Wakui, K. Shima, and M. Takahashi, «Analysis of saturated synchronous reactances of a large turbine generator by considering cross-magnetizing reactances using finite elements», *IEEE Transactions on Energy Conversion*, vol. 14, pp. 66–71, 1999. DOI: 10.1109/60.749149.

[47] C. Jacobina, E. R. C. Da Silva, A. Lima, and R. L. A. Ribeiro, «Vector and scalar control of a four switch three phase inverter», in *Industry Applications Conference (IAS)*, 1995. DOI: 10.1109/IAS.1995.530611.

[48] R. Kalman, «A new approach to linear filtering and prediction problems», *Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960. DOI: 10.1115/1.3662552.

[49] K. Kaur and S. Singh, «Analysis and comparison of scalar and vector control for adjustable speed ac drives», Master's thesis, Thapar University Patiala, India, 2011. [Online]. Available: http://dspace.thapar.edu:8080/dspace/handle/10266/1489.

[50] M. P. Kazmierkowski, R. Krishnan, J. D. Irwin, and F. Blaabjerg, *Control in Power Electronics: selected problems*. Academic Press, 2002, ISBN: 9780124027725.

[51] R. Kerkman, D. Leggate, D. Schlegel, and C. Winterhalter, «Effects of parasitics on the control of voltage source inverters», *IEEE Transactions on Power Electronics*, vol. 18, pp. 140–150, 2003. DOI: 10.1109/TPEL.2002.807190.

[52] E. Kerrigan and J. Maciejowski, «Invariant sets for constrained nonlinear discrete-time systems with application to feasibility in model predictive control», in *Conference on Decision and Control (CDC)*, 2000. DOI: 10.1109/CDC.2001.914717.

[53] H.-W. Kim, M.-J. Youn, K.-Y. Cho, and H.-S. Kim, «Nonlinearity estimation and compensation of pwm vsi for pmsm under resistance and flux linkage uncertainty», *IEEE Transactions on Control Systems Technology*, vol. 14, pp. 589–601, 2006. DOI: 10.1109/TCST.2006.876622.

[54] P. C. Krause, O. Wasynczuk, and S. Sudhoff, *Analysis of Electric Machinery and Drive Systems*. Wiley-IEEE Press, 2002, ISBN: 9781118024294.

[55] M. Kvasnica, P. Grieder, and M. Baotić. (2004). Multi-parametric toolbox (MPT), [Online]. Available: http://control.ee.ethz.ch/~mpt/.

[56] E. Levi, «Saturation modelling in d-q axis models of salient pole synchronous machine», *IEEE Transactions on Energy Conversion*, vol. 14, pp. 44–50, 1999. DOI: 10.1109/60.749146.

[57] A. Linder, R. Kanchan, R. Kennel, and P. Stolze, *Model-Based Predictive Control of Electric Drives*. Cuvillier Verlag Göttingen, 2010, ISBN: 9783869553986. [Online]. Available: http://www.eal.ei.tum.de/uploads/media/Model-based_predictive_control_of_electrical_drives.pdf.

[58] C. V. Loan, «Computing integrals involving the matrix exponential», *IEEE Transactions on Automatic Control*, vol. 23, pp. 395–404, 1978. DOI: 10.1109/TAC.1978.1101743.

[59] D. Luenberger and Y. Ye, *Linear and Nonlinear Programming*. Springer, 2008, ISBN: 9780387745039.

[60] D. Luenberger, «Observing the state of a linear system», *IEEE Transactions on Military Electronics*, vol. 8, pp. 74–80, 1964. DOI: 10.1109/TME.1964.4323124.

[61] J. M. Maciejowski, *Predictive Control with Constraints*. Prentice Hall, 2002, ISBN: 9780201398236.

[62] C. Mademlis and V. Agelidis, «On considering magnetic saturation with maximum torque per current control in interior permanent magnet synchronous motor drives», *IEEE Transactions on Energy Conversion*, vol. 16, pp. 246–252, 2001. DOI: 10.1109/60.937204.

[63] J. Mattingley and S. Boyd, «Cvxgen: a code generator for embedded convex optimization», *Optimization and Engineering*, vol. 13, pp. 1–27, 2012. DOI: 10.1007/s11081-011-9176-9. [Online]. Available: http://cvxgen.com/.

[64] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, «Constrained model predictive control: stability and optimality», *Automatica*, vol. 36, pp. 789–814, 2000. DOI: 10.1016/S0005-1098(99)00214-9.

[65] N. Mohan, T. M. Undeland, and W. P. Robbins, *Power Electronics*. John Wiley and Sons, 2003, ISBN: 9780471226932.

[66] L. Montejano, «Some results about minkowski addition and difference», *Mathematika*, vol. 43, pp. 265–273, 1996. DOI: 10.1112/S0025579300011761.

[67] Mosek. (2013). Mosek, [Online]. Available: http://mosek.com.

[68] D. Muñoz de la Peña and P. D. Christofides, «Lyapunov-based model predictive control of nonlinear systems subject to data losses», *IEEE Transactions on Automatic Control*, vol. 53, pp. 2076–2089, 2008. DOI: 10.1109/TAC.2008.929401.

[69] P. Mutschler, «A new speed-control method for induction motors», in *International Intelligent Motion Conference (PCIM)*, 1998. [Online]. Available: http://tubiblio.ulb.tu-darmstadt.de/10723.

[70] A. Nordvall, «Multilevel inverter topology survey», Master's thesis, Chalmers University of Technology, Sweden, 2011. [Online]. Available: http://publications.lib.chalmers.se/records/fulltext/173954/173954.pdf.

[71] J. Ojo and T. Lipo, «An improved model for saturated salient pole synchronous motors», *IEEE Transactions on Energy Conversion*, vol. 4, pp. 135–14, 1989. DOI: 10.1109/60.23164.

[72] A. Oliveira, C. Jacobina, and A. M. N. Lima, «Improved dead-time compensation for sinusoidal pwm inverters operating at high switching frequencies», *IEEE Transactions on Industrial Electronics*, vol. 54, pp. 2295–2304, 2007. DOI: `10.1109/TIE.2007.894770`.

[73] F. Parasliti and P. Poffet, «A model for saturation effects in high-field permanent magnet synchronous motors», *IEEE Transactions on Energy Conversion*, vol. 4, pp. 487–494, 1989. DOI: `10.1109/60.43253`.

[74] R. H. Park, «Two-reaction theory of synchronous machines generalized method of analysis - part i», *Transactions of the American Institute of Electrical Engineers*, vol. 48, pp. 716–727, 1929. DOI: `10.1109/T-AIEE.1929.5055275`.

[75] G. Pellegrino, P. Guglielmi, E. Armando, and R. Bojoi, «Self-commissioning algorithm for inverter nonlinearity compensation in sensorless induction motor drives», *IEEE Transactions on Industry Applications*, vol. 46, pp. 1416–1424, 2010. DOI: `10.1109/TIA.2010.2049554`.

[76] I. Pereira and A. Martins, «Experimental comparison of carrier and space vector pwm control methods for three-phase npc converters», in *International Conference on Renewable Energies and Power Quality (ICREPQ)*, 2009. [Online]. Available: `http://www.icrepq.com/ICREPQ'09/301-pereira.pdf`.

[77] K. B. Petersen and M. S. Pedersen, «The matrix cookbook», Technical University of Denmark, Tech. Rep., 2012. [Online]. Available: `http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=3274`.

[78] M. Preindl, «Switching frequency reduction using sensorless model predictive direct control for high power vsi», Master's thesis, ETH Zurich, 2009.

[79] M. Preindl and S. Bolognani, «Comparison of direct and pwm model predictive control for power electronic and drive systems», in *Applied Power Electronic Conference and Exposition (APEC)*, 2013. DOI: `10.1109/APEC.2013.6520651`.

[80] ——, «Model predictive direct speed control with finite control set of pmsm drive systems», *IEEE Transactions on Power Electronics*, vol. 28, pp. 1007–1015, 2013. DOI: `10.1109/TPEL.2012.2204277`.

[81] ——, «Model predictive direct torque control with finite control set for pmsm drive systems, part 1: maximum torque per ampere operation», *IEEE Transactions on Industrial Informatics*, vol. 9, pp. 1912–1921, 2013. DOI: `10.1109/TII.2012.2227265`.

[82] ——, «Model predictive direct torque control with finite control set for pmsm drive systems, part 2: field weakening operation», *IEEE Transactions on Industrial Informatics*, vol. 9, pp. 648–657, 2013. DOI: `10.1109/TII.2012.2220353`.

[83] M. Preindl, C. Danielson, and S. Bolognani, «Model predictive torque control with pwm using fast gradient method», in *Applied Power Electronic Conference and Exposition (APEC)*, 2013. DOI: 10.1109/APEC.2013.6520661.

[84] M. Preindl, E. Schaltz, and P. Thøgersen, «Switching frequency reduction using model predictive direct current control for high power voltage source inverters», *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 2826–2835, 2011. DOI: 10.1109/TIE.2010.2072894.

[85] S. Richter, C. N. Jones, and M. Morari, «Real-time input-constrained mpc using fast gradient methods», in *Conference on Decision and Control (CDC)*, 2009. DOI: 10.1109/CDC.2009.5400619.

[86] S. Richter, S. Mariéthoz, and M. Morari, «High-speed online mpc based on a fast gradient method applied to power converter control», in *American Control Conference (ACC)*, 2010.

[87] J. Rodríguez and P. Cortés, *Predictive Control of Power Converters and Electrical Drives*. John Wiley & Sons, 2012, ISBN: 9781119963981. DOI: 10.1002/9781119941446.

[88] D. E. Salt, D. Drury, D. Holliday, A. Griffo, P. Sangha, and A. Dinu, «Compensation of inverter nonlinear distortion effects for signal-injection-based sensorless control», *IEEE Transactions on Industry Applications*, vol. 47, pp. 2084–2092, 2011. DOI: 10.1109/TIA.2011.2161430.

[89] P. Sen, «Electric motor drives and control - past, present, and future», *IEEE Transactions on Industrial Electronics*, vol. 37, pp. 562–575, 1990. DOI: 10.1109/41.103462.

[90] A. El-Serafi and J. Wu, «Determination of the parameters representing the cross-magnetizing effect in saturated synchronous machines», *IEEE Transactions on Energy Conversion*, vol. 8, pp. 333–342, 1993. DOI: 10.1109/60.257042.

[91] E. D. Sontag, «A 'universal' construction of artstein's theorem on nonlinear stabilization», *Systems & Control Letters*, vol. 13, pp. 117–123, 1989. DOI: 10.1016/0167-6911(89)90028-5.

[92] W. Soong, S. Han, and T. Jahns, «Design of interior pm machines for field-weakening applications», in *International Conference on Electrical Machines and Systems (ICEMS)*, 2007.

[93] F. R. Stansel, «Some analyses of wave shapes used in harmonic producers», *Bell System Technical Journal*, vol. 20, pp. 331–339, 1941. [Online]. Available: http://www3.alcatel-lucent.com/bstj/vol20-1941/articles/bstj20-3-331.pdf.

[94] B. Stumberger, G. Stumberger, D. Dolinar, A. Hamler, and M. Trlep, «Evaluation of saturation and cross-magnetization effects in interior permanent-magnet synchronous motor», *IEEE Transactions on Industry Applications*, vol. 39, pp. 1264–1271, 2003. DOI: 10.1109/TIA.2003.816538.

[95] C. Sullivan, «Computationally efficient winding loss calculation with multiple windings, arbitrary waveforms, and two-dimensional or three-dimensional field geometry», *IEEE Transactions on Power Electronics*, vol. 16, pp. 142–150, 2001. DOI: 10.1109/63.903999.

[96] S.-A. Tahan and I. Kamwa, «A two-factor saturation model for synchronous machines with multiple rotor circuits», *IEEE Transactions on Energy Conversion*, vol. 10, pp. 609–616, 1995. DOI: 10.1109/60.475830.

[97] I. Takahashi and T. Noguchi, «A new quick-response and high-efficiency control strategy of an induction motor», *IEEE Transactions on Industry Applications*, vol. IA-22, pp. 820–827, 1986. DOI: 10.1109/TIA.1986.4504799.

[98] S.-J. Wang and S.-K. Lin, «Analytical prediction of the incremental inductance of the permanent magnet synchronous motors», *IEEE Transactions on Magnetics*, vol. 40, pp. 2044–2046, 2004. DOI: 10.1109/TMAG.2004.830629.

[99] Y. Wang and S. Boyd, «Fast model predictive control using online optimization», *IEEE Transactions on Control Systems Technology*, vol. 18, pp. 267–278, 2010. DOI: 10.1109/TCST.2009.2017934.

[100] D. White and H. Woodson, *Electromechanical Energy Conversion.* John Wiley & Sons, 1959, ISBN: 9780262230291.

[101] W. Wonham, «On the separation theorem of stochastic control», *SIAM Journal on Control*, vol. 6, pp. 312–326, 1968. DOI: 10.1137/0306023.

[102] K. Zhou, «Relationship between space-vector modulation and three-phase carrier-based pwm: a comprehensive analysis [three-phase inverters]», *IEEE Transactions on Industrial Electronics*, vol. 49, pp. 186–196, 2002. DOI: 10.1109/41.982262.

[103] M. Zordan, P. Vas, M. Rashed, S. Bolognani, and M. Zigliotto, «Field-weakening in high-performance pmsm drives: a comparative analysis», in *Industry Applications Conference (IAS)*, 2000. DOI: 10.1109/IAS.2000.882112.

List of Symbols

Sets, Matrices, Vectors

$\mathbb{N}$ ($\mathbb{N}_+$)	Set of integers (non-negative integers)
$\mathbb{R}$ ($\mathbb{R}_+$)	Set of real numbers (non-negative 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
$\mathbf{I}$	Identity matrix of appropriate dimension
$\mathbf{J}$	Rotation matrix $\mathbf{J} = [[0, -1]', [1, 0]']'$
$\mathbf{1}$	Vector of ones; $\mathbf{1} = [1, \ldots, 1]'$
$\mathbf{0}$	Vector of zeros; $\mathbf{0} = [0, \ldots, 0]'$

Algebraic Operators

$\mathbf{A}'$	Transpose of matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$
$\mathbf{A}^{\dagger}$	Moore-Penrose pseudoinverse of matrix $\mathbf{A}$
$\mathbf{A}_i$	i-th row of matrix $\mathbf{A}$
x_i	i-th element of vector x
$\operatorname{sgn} x$	Sign of x
$\lvert x \rvert$	Absolute value of x
$\lVert x \rVert_p$	p-norm of vector x; $\lVert x \rVert_p = (\sum \lvert x_i \rvert^p)^{1/p}$
$\lVert x \rVert$	Euclidean (or two) norm of vector x; $\lVert x \rVert = \sqrt{x'x}$
$\lVert x \rVert_2$	Squared Euclidean norm of vector x; $\lVert x \rVert_2 = \lVert x \rVert^2 = x'x$
$\lVert x \rVert_1$	Manhattan (or one) norm of vector x; $\lVert x \rVert_1 = \sum_{i=1}^{n} \lvert x_i \rvert$
$\lVert x \rVert_\infty$	Maximum (or infinity) norm of vector x; $\lVert x \rVert_\infty = \max_{i \in \{1,\ldots,n\}} \lvert x_i \rvert$

Set Operators

$\mathcal{P} \cap \mathcal{Q}$	Set intersection; $\mathcal{P} \cap \mathcal{Q} = \{x \in \mathbb{R}^n \mid x \in \mathcal{P} \text{ and } x \in \mathcal{Q}\}$
$\mathcal{P} \cup \mathcal{Q}$	Set intersection; $\mathcal{P} \cup \mathcal{Q} = \{x \in \mathbb{R}^n \mid x \in \mathcal{P} \text{ or } x \in \mathcal{Q}\}$
$\mathcal{P} \setminus \mathcal{Q}$	Set difference; $\mathcal{P} \setminus \mathcal{Q} = \{x \in \mathbb{R}^n \mid x \in \mathcal{P} \text{ and } x \notin \mathcal{Q}\}$
$\mathcal{P} \oplus \mathcal{Q}$	Minkowski sum; $\mathcal{P} \oplus \mathcal{Q} = \{x + y \in \mathbb{R}^n \mid x \in \mathcal{P},\ y \in \mathcal{Q}\}$
$\mathcal{P} \ominus \mathcal{Q}$	Pontryagin difference; $\mathcal{P} \ominus \mathcal{Q} = \{x \in \mathbb{R}^n \mid x + y \in \mathcal{P},\ \forall y \in \mathcal{Q}\}$
$\mathcal{P} \subset (\subseteq) \mathcal{Q}$	$\mathcal{P}$ is a subset (strict subset) of $\mathcal{Q}$
$\operatorname{br} \mathcal{P}$	Boundary of $\mathcal{P}$
$\operatorname{int} \mathcal{P}$	Interior of $\mathcal{P}$; $\operatorname{int} \mathcal{P} = \mathcal{P} \setminus \operatorname{br} \mathcal{P}$
$\operatorname{hull} \mathcal{P}$	Convex hull of $\mathcal{P}$
$\operatorname{ball} \mathcal{P}$	Chebyshev ball of $\mathcal{P}$

Acronyms

AC	Alternating current
CCS	Convex control set
CFTOC	Constrained finite time optimal control
CLF	Control Lyapunov function
DC	Direct current
DSVM	Discontinuous SVM
FCS	Finite control set
FW	Field weakening
lp	Linear programming
MPC	Model predictive control
MTPA	Maximum torque per ampere
PM	Permanent magnet
PMSM	Permanent magnet synchronous machine
PWM	Pulse width modulation
qp	Quadratic programming
RHC	Receding horizon control
SSVM	Symmetric SVM
SVM	Space vector modulation
VSI	Voltage source inverter