

# Model Predictive Control for Power Electronics Applications

Daniel E. Quevedo, Ricardo P. Aguilera, and Tobias Geyer

**Abstract** Power electronics converters use switching elements to manipulate voltage and current waveforms. This enables the interconnection of components having different requirements, e.g., when incorporating renewable energy sources into the grid. The use of switching elements may lead to high energy efficiency. However, switching dynamical systems are difficult to analyse and design. In this chapter, we outline how model predictive control concepts can be used in power electronics and electrical drives. Special emphasis is given on the finite-set nature of manipulated variables and associated stability and optimisation issues. For particular classes of system models, we discuss practical algorithms, which make long-horizon predictive control suitable for power electronics applications.

## 1 Introduction

Advances in the field of power electronics allow engineers to manipulate electrical power and to control its flow efficiently with power levels ranging from milliwatt to gigawatt. The utilization of power electronics has increased considerably in recent years. In 2015, the overall market size was USD 36 billion [42]. The power electronics market can be divided into industrial applications, utility-scale power electronics [13], automotive [14], consumer electronics, aerospace and defense, and

---

Daniel E. Quevedo  
Chair for Automatic Control (EIM-E), Paderborn University, 33098 Paderborn, Germany, e-mail: dquevedo@ieee.org

Ricardo P. Aguilera  
School of Electrical, Mechanical and Mechatronic Systems, University of Technology Sydney, NSW, 2007, Australia, e-mail: raguilera@ieee.org

Tobias Geyer  
ABB Corporate Research, ABB Switzerland Ltd, Power Electronic Systems, Segelhofstrasse 1 K, 5405 Baden-Dättwil, Switzerland, e-mail: t.geyer@ieee.org

information and communication technology. Notable examples of industrial applications include renewable energy systems [8], rail traction and motor drives [15]. Power converters have been constantly advanced regarding their semiconductors, packaging, passive materials, topologies and control techniques [30].

From a control systems perspective, power electronic systems give rise to intrinsically challenging design problems. Specifically, three major challenges can be identified:

1. **Switched dynamics.** The main building blocks of power electronic systems are linear circuit elements, such as inductors, capacitors and resistors, which are complemented by semiconductor switches. The latter are either actively controlled or (passive) diodes. As a result, when controlling currents, fluxes and voltages and manipulating the switch positions, power electronic systems constitute switched *linear* systems, provided that saturation effects of magnetic material, delays and safety constraints can be neglected [21, 53].  
In general, however, power electronic systems represent switched *nonlinear* systems. Nonlinearities arise, for example, when machine variables such as the electromagnetic torque or stator flux magnitude are directly controlled; both quantities are nonlinear functions of currents or flux linkages. For grid-connected converters, the real and reactive power is nonlinear in terms of the currents and voltages. Saturation effects in inductors and current constraints lead to additional nonlinearities.
2. **MIMO systems.** Three-phase power converters have at least three manipulated variables, i.e., one switch position per phase. In the simplest case, the current of an inductive load needs to be controlled. When the star point of the load floats, two linearly independent currents arise, resulting in a system with two controlled variables and three manipulated variables. For more complicated systems, such as converters with *LC* filters and inductive loads, six controlled variables result. Dc-ac modular multilevel converters (MMC) [36] with  $n$  modules per arm are significantly more complex with up to  $6n$  manipulated variables and up to  $6n + 6$  controlled variables.
3. **Short computation times.** The third challenge results from the short sampling intervals of 1 ms and less that are typically used in power electronic systems. These short sampling intervals limit the time available to compute the control actions. To reduce the cost of power electronic converters sold in high volumes, cheap computational hardware is usually deployed as the control platform. Replacing existing control loops with only low computational requirements by new and computationally more demanding methods exasperates the challenge of short sampling intervals. This is particularly the case for direct control methods that avoid the use of a modulator. These methods typically require very short sampling in the range of 25  $\mu$ s.

To address these challenges, various embodiments of model predictive control (MPC) principles have emerged as a promising control alternative for power conversion applications [9, 20, 31, 50, 51, 54]. As we shall see in this chapter, this pop-

ularity of MPC is due to the fact that predictive control algorithms present several advantages that make them suitable for the control of power electronic systems:

1. The concepts are intuitive and easy to understand;
2. MPC can handle converters with multiple switches and states, e.g., current, voltage, power, torque, etc.;
3. constraints and nonlinearities can be easily included; and
4. the resulting controller is, in general, easy to implement.

## 2 Basic Concepts

Various MPC methods have been proposed for controlling power electronic systems. Here, one can distinguish between formulations that use system models governed by linear time-invariant dynamics, and those that incorporate nonlinearities. Most MPC strategies are formulated in a discrete-time setting with a fixed sampling interval, say  $h > 0$ . System inputs are restricted to change their values only at the discrete sampling instants, i.e., at times  $t = kh$ , where  $k \in \mathbb{N} \triangleq \{0, 1, 2, \dots\}$  denotes the sampling instants.

Since power electronics applications are often governed by nonlinear dynamic relations, it is convenient to represent the system to be controlled in discrete-time state space form via:

$$x(k+1) = f(x(k), u(k)), \quad k \in \mathbb{N}, \quad (1)$$

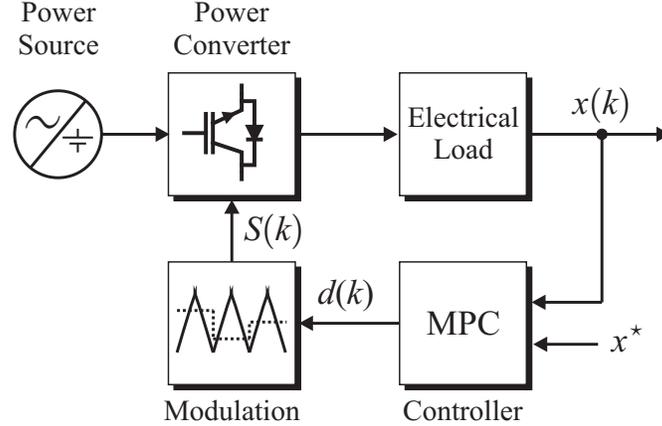
where  $x(k) \in \mathbb{R}^n$  denotes the state value at time  $k$  and  $u(k) \in \mathbb{R}^m$  is the plant input. Depending on the application at hand, the system state is a vector, which may contain capacitor voltages, inductor and load currents, and fluxes.

### 2.1 System Constraints

An interesting feature of the MPC framework is that it allows one to incorporate state and input constraints, say:

$$\begin{aligned} x(k) &\in \mathbb{X} \subseteq \mathbb{R}^n, & k &\in \{0, 1, 2, \dots\}, \\ u(k) &\in \mathbb{U} \subseteq \mathbb{R}^m, & k &\in \{0, 1, 2, \dots\}. \end{aligned} \quad (2)$$

State constraints can, for example, correspond to constraints on capacitor voltages in flying capacitor converters or neutral point clamped converters. Constraints on load currents can also be modeled as state constraints. Throughout this chapter we will focus on input constraints, since they naturally arise when controlling power converters.



**Fig. 1** MPC with continuous control set.

Input constraints,  $u(k) \in \mathbb{U}$ , are related to the switch positions during the interval  $(kh, (k+1)h]$ . If a modulator is used, then  $u(k)$  will be constrained to belong to a bounded continuous set. For example, the components of  $u(k)$  could correspond to duty cycles,  $d(k)$ , or PWM reference signals. In this case, the control input is constrained by

$$u(k) = d(k) \in \mathbb{U} \triangleq [-1, 1]^m \subset \mathbb{R}^m, \quad k \in \{0, 1, 2, \dots\}, \quad (3)$$

where  $m$  denotes the number of phases, see Fig. 1. Clearly, the above model can only approximate switching effects, see also [35]. Nevertheless, as we will see, several interesting and powerful controllers for power converters have been developed by using this simple setting.

On the other hand, in so-called *direct control* applications, where no modulator is used,  $u(k)$  is constrained to belong to a finite set describing the available switch combinations. Such approaches have attracted significant attention in the power electronics community, often under term finite control set MPC (FCS-MPC) [51]. The main advantage of this predictive control strategy comes from the fact that switching actions, say  $S(k)$ , are directly taken into account in the optimization procedure as constraints on the system inputs, see Fig. 2. Thus, the control input is restricted to belong to a finite set represented by

$$u(k) = S(k) \in \mathbb{U} \subset \mathbb{R}^m, \quad k \in \{0, 1, 2, \dots\}, \quad (4)$$

where  $\mathbb{U}$  is an integer set obtained by combining the  $m$  switch values. For the control of multilevel topologies, it is often convenient to consider the resultant phase voltage level as the control input rather than the switch position of each semiconductor switch. For example, for a five-level inverter,  $\mathbb{U} = \{-2, -1, 0, 1, 2\}^m$ .

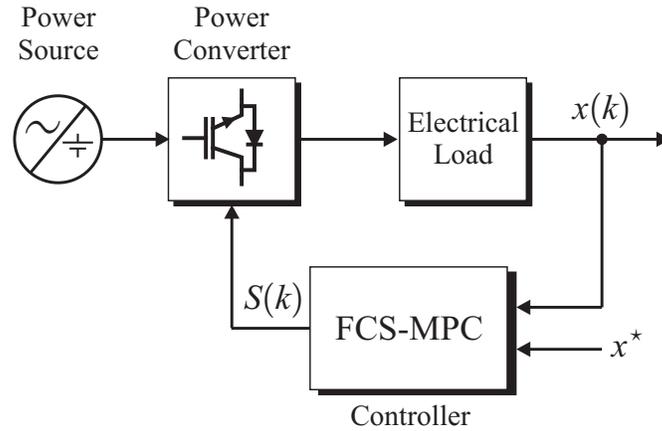


Fig. 2 MPC with finite control set (direct control).

## 2.2 Cost Function

A distinguishing element of MPC, when compared to other control algorithms, is that at each time instant  $k$  and for a given (measured or estimated) plant state  $x(k)$ , a cost function over a finite horizon of length  $N$  is minimized. The following choice encompasses many alternatives documented in the literature:

$$V(x(k), \mathbf{u}'(k)) \triangleq F(x'(k+N)) + \sum_{\ell=k}^{k+N-1} L(x'(\ell), u'(\ell)). \quad (5)$$

Here,  $L(\cdot, \cdot)$  and  $F(\cdot)$  are weighting functions, which serve to penalize predicted system behaviour, e.g., differences between references for voltages and currents and their predicted values, see Sect. 2.4.

For example, for a two-level three-phase inverter in orthogonal  $\alpha\beta$  coordinates, one can use (see [52])

$$L(x'(\ell), u'(\ell)) = \lambda_1 (i_\alpha(\ell) - i_\alpha^*)^2 + \lambda_2 (i_\beta(\ell) - i_\beta^*)^2.$$

For a one-phase three-cell flying capacitor converter (FCC) one can choose (see, e.g., [37])

$$L(x'(\ell), u'(\ell)) = \lambda_1 (i_a(\ell) - i_a^*)^2 + \lambda_2 (v_{c1}(\ell) - v_{c1}^*)^2 + \lambda_3 (v_{c2}(\ell) - v_{c2}^*)^2.$$

In (5), predicted plant state values,  $x'(\ell)$ , are formed using the system model (1):

$$x'(\ell+1) = f(x'(\ell), u'(\ell)), \quad \ell \in \{k, k+1, \dots, k+N-1\} \quad (6)$$

where

$$u'(\ell) \in \mathbb{U}, \quad \ell \in \{k, k+1, \dots, k+N-1\}$$

refers to tentative plant inputs (to be decided). The recursion (6) is initialized with the current plant state measurement (or estimate), i.e.:

$$x'(k) \leftarrow x(k). \quad (7)$$

Thus, (6) refers to predictions of the plant states that would result if the plant inputs at the update times  $\{k, k+1, \dots, k+N-1\}$  were set equal to the corresponding values in

$$\mathbf{u}'(k) \triangleq [u'^T(k) u'^T(k+1) \dots u'^T(k+N-1)]^T. \quad (8)$$

Both, the predicted plant state trajectory and the plant inputs are constrained in accordance with (2), i.e., we have:

$$\begin{aligned} u'(\ell) &\in \mathbb{U}, \quad \forall \ell \in \{k, k+1, \dots, k+N-1\} \\ x'(\ell) &\in \mathbb{X}, \quad \forall \ell \in \{k+1, k+2, \dots, k+N\}. \end{aligned}$$

Constrained minimization of  $V(\cdot, \cdot)$  in (5) gives the optimizing control sequence at time  $k$  and for state  $x(k)$ :

$$\mathbf{u}^{\text{opt}}(k) \triangleq [(u^{\text{opt}}(k))^T (u^{\text{opt}}(k+1;k))^T \dots (u^{\text{opt}}(k+N-1;k))^T]^T. \quad (9)$$

It is worth emphasizing here that, in general, plant state predictions,  $x'(\ell)$ , will differ from actual plant state trajectories,  $x(\ell)$ . This is a consequence of possible model inaccuracies and the moving horizon optimization paradigm described next.

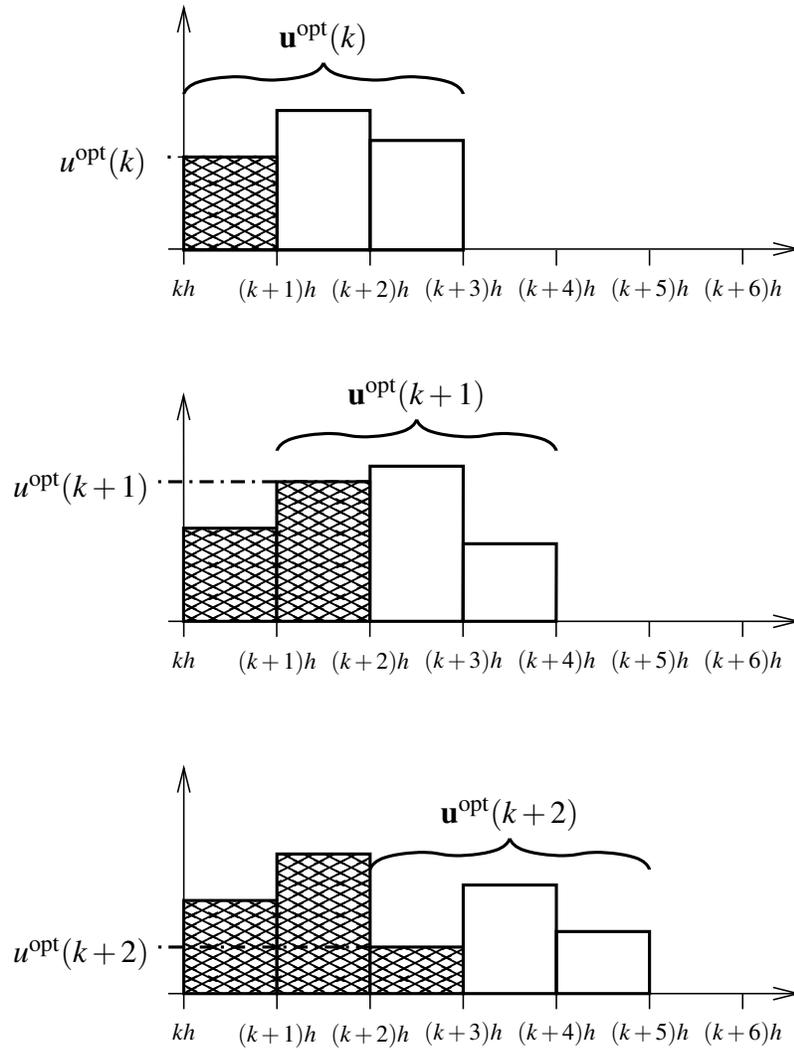
### 2.3 Moving Horizon Optimization

Despite the fact that the optimizer  $\mathbf{u}^{\text{opt}}(k)$  in (9) contains feasible plant inputs over the entire horizon,  $(kh, (k+N-1)h]$ , in most MPC approaches, only the first element is used, i.e., the system input in (1) is set to

$$u(k) \leftarrow u^{\text{opt}}(k).$$

At the next sampling step, i.e., at discrete-time  $k+1$ , the system state  $x(k+1)$  is measured (or estimated), the horizon is shifted by one step, and another optimization is carried out. This yields  $\mathbf{u}^{\text{opt}}(k+1)$  and its first element provides  $u(k+1) = u^{\text{opt}}(k+1)$ , etc. As illustrated in Fig. 3 for a horizon length  $N=3$ , the horizon taken into account in the minimization of the cost function  $V$  slides forward as  $k$  increases.

The design of observers for the system state is beyond the scope of this chapter. The interested reader is referred to [2, 17, 22], which illustrate the use of Kalman filters for MPC formulations in power electronics.



**Fig. 3** Moving horizon principle with horizon length  $N = 3$ .

## 2.4 Design Parameters

As seen above, MPC allows one to treat multi-variable nonlinear systems in an, at least conceptually, simple way. In addition to choosing the sampling interval  $h$  (which, amongst other things, determines the system model (1)), MPC design essentially amounts to selecting the cost function, i.e., the weighting functions  $F(\cdot)$  and  $L(\cdot, \cdot)$ , and the horizon length  $N$ .

As we shall see, the design of the weighting functions should take into account the actual control objectives and may also consider stability issues [3, 43]).<sup>1</sup> For example, tracking of the desired output and internal voltages and currents (which are assumed to be given, cf., [46]) can be accommodated into the MPC framework by choosing weights that penalize a measure of the difference between predicted and reference values.

For a given sampling frequency  $1/h$ , larger values for the horizon length  $N$  will in general provide better performance, as quantified by the weighting functions  $F(\cdot)$  and  $L(\cdot, \cdot)$ . Indeed, one can expect that, for large enough  $N$ , the effect of  $u(k)$  on  $x'(\ell)$  for  $\ell > k + N$  will be negligible and, consequently, MPC will approximate the performance of an infinite horizon optimal controller [27, 45]. On the other hand, the constrained optimization problem which, in principle, needs to be solved on-line to find the controller output, has a computational complexity which, in general, increases with the horizon length. As a consequence, the horizon parameter  $N$  allows the designer to trade-off performance versus on-line computational effort.

### 3 Linear Quadratic MPC for Converters with a Modulator

Most power converters use a modulation stage to synthesise the switching signals. To simplify the design of control strategies, it is common practice to separate control and modulation issues, see, e.g., [29]. By averaging the switching signal, the switching nature of the power converter can be concealed, provided that the switching frequency per fundamental frequency is high, a modulation method with a fixed modulation cycle is used and sampling is performed when the voltage and current ripples due to modulation are close to zero. If these conditions are fulfilled, one may use standard methods for the controller design. As we shall see below, for the case of MPC, the situation is similar.

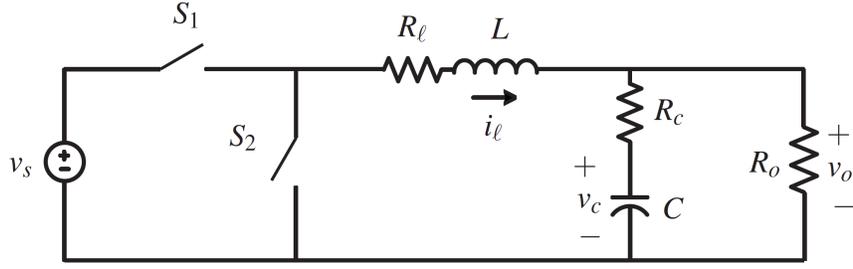
A particularly simple case of (5)–(6) arises when the cost function is quadratic and the system model is linear and time-invariant, i.e.:

$$\begin{aligned} V(x(k), \mathbf{u}'(k)) &= x'^T(k+N)Px'(k+N) + \sum_{\ell=k}^{k+N-1} \left\{ x'^T(\ell)Qx'(\ell) + u'^T(\ell)Ru'(\ell) \right\}, \\ x'(\ell+1) &= Ax'(\ell) + Bu'(\ell), \\ x'(\ell) &\in \mathbb{X} \subseteq \mathbb{R}^n, \quad u'(\ell) \in \mathbb{U} \subseteq \mathbb{R}^m, \quad \ell \in \{k, k+1, \dots, k+N-1\}, \end{aligned} \tag{10}$$

where  $A$  and  $B$  denote the state-update and input matrices, and  $P$ ,  $Q$  and  $R$  are positive semi-definite matrices of appropriate dimensions. The constraint sets  $\mathbb{X}$  and  $\mathbb{U}$  are polyhedra.

---

<sup>1</sup> Note that the weighting functions should be chosen such that  $V(\cdot, \cdot)$  depends on the decision variables contained in  $\mathbf{u}'(k)$ , see (8).



**Fig. 4** Topology of the dc-dc buck converter

Despite the ever growing computational power available and recent advances in implementing quadratic programming (QP) solvers on embedded system architectures, solving the QP in real-time for power electronics applications poses a highly challenging problem. When using sampling intervals in the  $\mu s$  range, the computation times needed to solve the QP typically exceed the sampling interval—often by one or two orders of magnitude. Rather than solving the mathematical optimization problem in real-time for the given state vector at the current time-step, the optimization problem can be solved offline for *all possible* states. Specifically, so-called (explicit) *state-feedback control laws* as presented in previous parts of this book, can be computed for all states  $x(k) \in \mathbb{X}$  [6]. Explicit control laws are characterised via a polyhedral partition of the state space which can be stored in a look-up table. The optimal control input can thus be read from the look-up table in a computationally efficient manner.

*Example 1.* To further illustrate the derivation and properties of the explicit state-feedback control law of MPC, consider a dc-dc step-down synchronous converter. The latter is commonly referred to as a buck converter, and it is shown in Fig. 4. Using the classic technique of averaging between the on and off modes of the circuit, the discrete-time system model

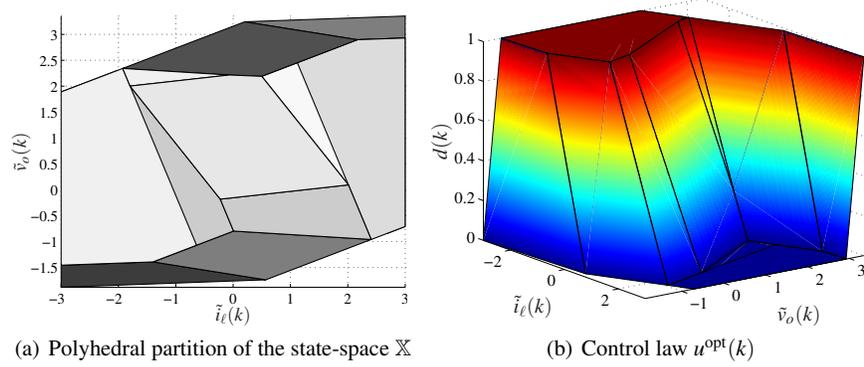
$$x(k+1) = Ax(k) + Bv_s d(k) \quad (11)$$

can be obtained, where  $v_s$  denotes the unregulated input voltage and  $d(k)$  the duty cycle. The state vector contains the inductor current  $i_l$  and the output voltage  $v_o$ , i.e.  $x = [i_l \ v_o]^T$ . From Fig. 4, the continuous-time system matrices are

$$F = \begin{bmatrix} -R_l/L & -1/L \\ \frac{R_o}{R_o+R_c} \frac{L-R_c R_l C}{LC} & -\frac{1}{R_o+R_c} \frac{L+R_c R_o C}{LC} \end{bmatrix}, \quad G = \begin{bmatrix} 1/L \\ \frac{R_o}{R_o+R_c} \frac{R_c}{L} \end{bmatrix}, \quad (12)$$

whereas their discrete-time representations in (11) are given by

$$A = e^{Fh}, \quad B = \int_0^h e^{F\tau} G d\tau. \quad (13)$$



**Fig. 5** Explicit state-feedback control law for the dc-dc buck converter over the state-space  $\mathbb{X}$  spanned by the scaled inductor current  $\tilde{i}_\ell(k)$  and the scaled output voltage  $\tilde{v}_o(k)$

Adopting the per unit (pu) system, the parameters in (12) are here taken as the inductor  $L = 3$  pu, capacitor  $C = 20$  pu and output resistor  $R_o = 1$  pu. The internal resistor of the inductor is set to  $R_\ell = 0.05$  pu and the equivalent series resistance of the capacitor is  $R_c = 0.005$  pu. The nominal input voltage is assumed to be  $v_s = 1.8$  pu.

To allow for variations in the input voltage, it is convenient to scale the system equations by  $v_s$ , as proposed in [22]. To this end, we define  $\tilde{i}_\ell = i_\ell/v_s$ ,  $\tilde{v}_o = v_o/v_s$  and  $\tilde{\mathbf{x}} = [\tilde{i}_\ell \ \tilde{v}_o]^T$ , and rewrite (11) as

$$\tilde{\mathbf{x}}(k+1) = A\tilde{\mathbf{x}}(k) + Bd(k). \quad (14)$$

Note that, unlike (11), (14) is linear in the state vector and the duty cycle.

The control objective is to regulate the output voltage to its reference  $v_o^*$  and to maintain the inductor current below its maximal allowed limit  $i_{\ell,\max}$  by manipulating the duty cycle. The latter is bounded between zero and one. This control problem can be captured by the optimization problem (cf., (10))

$$\begin{aligned} V(\tilde{\mathbf{x}}(k), \mathbf{u}(k)) &= \sum_{\ell=k}^{k+N-1} \left\{ (\tilde{\mathbf{x}}'(\ell) - \tilde{\mathbf{x}}^*)^T Q (\tilde{\mathbf{x}}'(\ell) - \tilde{\mathbf{x}}^*) + R(u'(\ell))^2 \right\}, \\ \tilde{\mathbf{x}}'(\ell+1) &= A\tilde{\mathbf{x}}'(\ell) + B\mathbf{u}'(\ell), \\ \tilde{\mathbf{x}}'(\ell) &\in \mathbb{X}, \mathbf{u}'(\ell) \in \mathbb{U}, \quad \ell \in \{k, k+1, \dots, k+N-1\}, \end{aligned} \quad (15)$$

where we set  $Q = \text{diag}(0, 1)$ ,  $R = 0.1$ ,  $\mathbb{X} = [-\tilde{i}_{\ell,\max}, \tilde{i}_{\ell,\max}] \times [-10, 10]$  and  $\mathbb{U} = [0, 1]$ . Note that  $\tilde{i}_{\ell,\max} = i_{\ell,\max}/v_s$  and  $u = d$ . To facilitate the regulation of the output voltage to a non-zero reference, we define  $\tilde{\mathbf{x}}^* = [0 \ \tilde{v}_o^*]^T$  with  $\tilde{v}_o^* = v_o^*/v_s$ . We assume  $\tilde{v}_o^* = 0.5$  and choose the horizon  $N = 3$ .

The explicit control law can be computed using the MPT toolbox [33]. The two-dimensional state-space is partitioned into 20 polyhedra. Using optimal complexity

reduction [25], an equivalent control law with 11 polyhedra can be derived, as shown in Fig. 5(a). The corresponding state-feedback controller providing  $u(k) = u^{\text{opt}}(k)$  is shown in Fig. 5(b). Note that the duty cycle is limited by zero and one as a result of the design procedure. An additional patch, such as an anti-windup scheme, is not required, see also [12]. ■

A similar MPC scheme was proposed in [39]. This rather basic controller can be enhanced in various ways. In the context of dc-dc converters, it is usually preferred to penalize the *change* in the duty cycle rather than the duty cycle as such, by introducing  $\Delta u(k) = u(k) - u(k-1)$  and penalizing  $R(\Delta u(\ell))^2$  rather than  $R(u(\ell))^2$  in (15). To enhance the voltage regulation at steady-state by removing any dc offset, an integrator state can be added [39]. Load variations can be addressed by a Kalman filter, see [22].

In the context of power electronics and drives applications, such MPC formulations have been studied extensively. One of the earliest references is [38], which proposes an explicit MPC controller in a field-oriented controller setting for an electrical drive. These initial results are extended in [40]. In [7], the speed and current control problem of a permanent-magnet synchronous machine is solved using MPC. Drives with flexible shafts are considered in [11], whereas [41] focuses on active rectifier units with LC filters.

## 4 Linear Quadratic Finite Control Set MPC

Controlling power converters without a modulator has received significant interest in recent years, leading to *direct control* methods. These methods combine the inner control loop, which typically controls the load currents, and the modulator in one computational stage. In doing so, the intrinsic delay of the modulator is avoided and the switching nature of the power converter can be directly addressed.

One of the most popular predictive control strategy for power electronic systems is FCS-MPC [9, 51]. This predictive control strategy explicitly models the switch positions by means of a finite control set. This implies that the input constraint set has a finite number of elements, as, for example, in (4).

In general, large prediction horizons  $N$  are preferable when using MPC. However, finding the optimal input sequence in case of FCS-MPC typically requires one to solve a combinatorial optimization problem [48]. Interestingly, for some topologies, one-step horizon MPC provides already good closed-loop performance [31, 50].

### 4.1 Closed-Form Solution

Consider again a quadratic cost function and a linear time-invariant system model. Unlike as in (10), however, the input constraint set  $\mathbb{U}$  is now a finite control set. Here-

after, we revisit the closed-form expression for the solution to this linear quadratic FCS-MPC problem, as presented in [26, 48].

Firstly, we define the predicted state sequence

$$\mathbf{x}'_{[1:N]}(k) \triangleq [x'^T(k+1) \ x'^T(k+2) \ \dots \ x'^T(k+N)]^T. \quad (16)$$

The subscript  $_{[1:N]}$  indicates that, unlike in (8), the state sequence is shifted by one time step.

Considering an initial system state  $x'(k) = x(k)$ , see also (7), we obtain

$$\mathbf{x}'_{[1:N]}(k) = \Phi \mathbf{u}'(k) + \Lambda x'(k),$$

where

$$\Phi \triangleq \begin{bmatrix} B & 0 & \dots & 0 & 0 \\ AB & B & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A^{N-1}B & A^{N-2} & \dots & AB & B \end{bmatrix}, \quad \Lambda \triangleq \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}.$$

In the following, we drop the time dependence of the state and input sequences in order to simplify the notation. The cost function (10) can then be re-written as

$$V(x, \mathbf{u}') = v(x) + \mathbf{u}'^T W \mathbf{u}' + 2\mathbf{u}'^T F x, \quad (17)$$

where  $x = x(k)$ ,  $\mathbf{u}' = \mathbf{u}'(k)$  and the term  $v(x)$  is independent of  $\mathbf{u}'$ . In (17),

$$\begin{aligned} W &\triangleq \Phi^T \mathcal{Q} \Phi + \mathcal{R} \in \mathbb{R}^{Nm \times Nm}, \\ F &\triangleq \Phi^T \mathcal{Q} \Lambda \in \mathbb{R}^{Nm \times n}, \end{aligned}$$

with

$$\begin{aligned} \mathcal{Q} &\triangleq \text{diag}\{Q, \dots, Q, P\} \in \mathbb{R}^{Nn \times Nn}, \\ \mathcal{R} &\triangleq \text{diag}\{R, \dots, R\} \in \mathbb{R}^{Nm \times Nm}. \end{aligned}$$

Notice that, if  $Q$  and  $R$  are positive definite, so is  $W$ .

**Remark 1 (Unconstrained Solution)** *If system constraints are not taken into account, i.e.  $\mathbb{U} \triangleq \mathbb{R}^m$  and  $\mathbb{X} \triangleq \mathbb{R}^n$ , then  $V(x, \mathbf{u}')$  is minimized when*

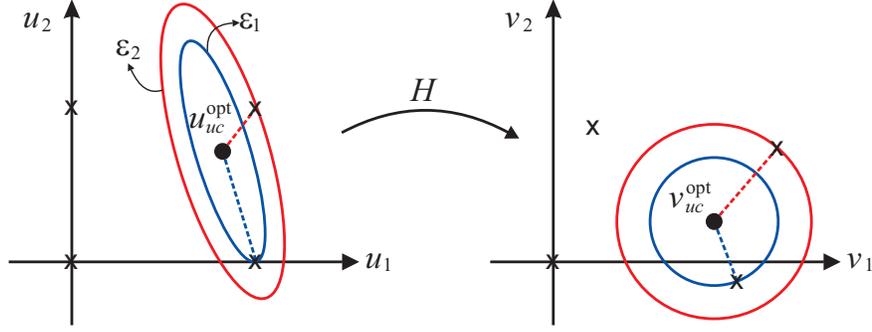
$$\mathbf{u}'_{uc}{}^{\text{opt}}(x) \triangleq \arg \left\{ \min_{\mathbf{u}' \in \mathbb{R}^{Nm}} V(x, \mathbf{u}') \right\} \triangleq -W^{-1} F x. \quad (18)$$

Based on the unconstrained optimum, it is convenient to rewrite the cost function (17) as:

$$V(x, \mathbf{u}') = (\mathbf{u}' - \mathbf{u}'_{uc}{}^{\text{opt}}(x))^T W (\mathbf{u}' - \mathbf{u}'_{uc}{}^{\text{opt}}(x)) + g(x), \quad (19)$$

where the term  $g(x)$  is independent of  $\mathbf{u}'$ .

To obtain the optimal finite set constrained solution one must find the control input which minimizes  $V(x, \mathbf{u}')$ . From (19), it follows that level sets of the cost func-



**Fig. 6** Geometrical representation of the optimal solution for FCS-MPC with  $u_1, u_2 \in \{0, 1\}$  and the horizon  $N = 1$ .

tion are ellipsoids, where the eigenvectors of  $W$  define the principal directions of the ellipsoid. Thus, the constrained optimizer  $\mathbf{u}^{\text{opt}}(x)$  does not necessarily correspond to the nearest neighbor of  $\mathbf{u}_{uc}^{\text{opt}}(x)$  within the constraint set  $\mathbb{U}^N$ .

*Example 2.* Consider the case where a power converter, modeled as a linear time-invariant model, has two semiconductor switches, which can take two values, i.e.,  $u_1, u_2 \in \{0, 1\}$ . Thus, the control input belongs to the following finite set:

$$u \in \mathbb{U} \triangleq \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\} \subset \mathbb{R}^2. \quad (20)$$

A geometrical representation of the situation for the case when the horizon is  $N = 1$  is depicted in Fig. 6 (left). Here, the ellipses,  $\varepsilon_i$  centered in  $u_{uc}^{\text{opt}}$ , represent all the points that lead to the same cost. Formally, if  $a, b \in \varepsilon_i$  then,  $V(x, a) = V(x, b)$ .

As we move away from the centre, the ellipses become larger, increasing the cost function value, i.e., if  $a \in \varepsilon_1$  and  $b \in \varepsilon_2$  then,  $V(x, a) < V(x, b)$ . Thus, in this example, the optimal solution, which produces the minimum cost function value is  $u^{\text{opt}} = [1 \ 0]^T$ , despite the nearest vector to the unconstrained solution being  $u = [1 \ 1]^T$ . Clearly, the optimal solution is, in general, not the nearest neighbour to the unconstrained solution. ■

Based on the above observations, one can derive a closed-form solution to the finite-set constrained optimisation problem at hand.

**Definition 1 (Vector Quantizer (see e.g. [18]))** Consider a set  $\mathcal{A} \subseteq \mathbb{R}^n$  and a countable (not necessarily finite) set  $\mathcal{B} \triangleq \{b_i\} \subset \mathbb{R}^n$ ,  $i \in \mathcal{I} \subseteq \mathbb{N}$  which satisfies that  $\exists \varepsilon > 0 : |b_i - b_j| \geq \varepsilon, \forall i, j \in \mathcal{I}$ . A function  $q_{\mathcal{B}}(\cdot) : \mathcal{A} \rightarrow \mathcal{B}$  is an Euclidean vector quantizer if  $q_{\mathcal{B}}(a) = b_i \in \mathcal{B}$  if and only if  $b_i$  satisfies that  $|a - b_i| \leq |a - b_j|$ , for all  $b_j \neq b_i$ , where  $b_j \in \mathcal{B}$ . The associated quantization error is defined as  $\bar{\eta}_{\mathcal{B}}(a) \triangleq q_{\mathcal{B}}(a) - a$ .

**Theorem 1 ([48])** Denote the elements of  $\mathbb{U}^N \triangleq \mathbb{U} \times \dots \times \mathbb{U}$  via  $\{\mu_1, \dots, \mu_r\}$ . Consider a matrix  $H$  that satisfies  $H^T H = W$ . Then, the constrained optimizer

$$\mathbf{u}^{opt}(x) \triangleq \arg \left\{ \min_{\mathbf{u}' \in \mathbb{U}^N} V(x, \mathbf{u}') \right\} \quad (21)$$

is given by

$$\mathbf{u}^{opt}(x) = H^{-1} q_{\mathbb{V}} \left( H^{-1} \mathbf{u}_{uc}^{opt}(x) \right) = H q_{\mathbb{V}} \left( -H^{-T} Fx \right), \quad (22)$$

where the vector quantizer  $q_{\mathbb{V}}$  maps  $\mathbb{R}^{Nm}$  to  $\mathbb{V}$ . The latter set is defined via  $\mathbb{V} \triangleq \{\mathbf{v}_1, \dots, \mathbf{v}_r\} \subset \mathbb{R}^{Nm}$ , in which  $\mathbf{v}_i = H\boldsymbol{\mu}_i$  for all  $\boldsymbol{\mu}_i \in \mathbb{U}^N$ .

*Proof.* To obtain the optimal solution, we define  $\mathbf{v}' = H\mathbf{u}'$ . Now, the cost function (19) can be expressed as:

$$V(x, \mathbf{v}') \triangleq (\mathbf{v}' - \mathbf{v}_{uc}^{opt}(x))^T (\mathbf{v}' - \mathbf{v}_{uc}^{opt}(x)) + g(x), \quad (23)$$

where

$$\mathbf{v}_{uc}^{opt}(k) \triangleq H\mathbf{u}_{uc}^{opt}(x).$$

Thus, in terms of  $\mathbf{v}'$ , the level sets of the cost function describe spheres centered at  $\mathbf{v}_{uc}^{opt}$ , as depicted in Fig. 6 (right). Therefore, in terms of these transformed variables, the nearest vector to the unconstrained solution,  $\mathbf{v}_{uc}^{opt}(x)$  is indeed the (constrained) optimal solution. ■

Notice that since  $W$  is symmetric and positive definite, then it is always possible to obtain a matrix  $H$  that satisfies  $H^T H = W$ , e.g.,  $H = W^{1/2}$ , as chosen in [48].

## 4.2 Design for Stability and Performance

We will next investigate stabilizing properties of FCS-MPC. For that purpose, we will include additional terminal constraints in the problem formulation of Section 2.2. This will allow us to adapt robust control concepts to suit the problem at hand.

For our subsequent analysis, we shall assume that the pair  $(A, B)$  is stabilizable and that the matrices  $Q$  and  $R$  are positive definite. A widely-used idea to establishing stability of MPC is based on finding a known control policy, say  $\kappa_f(x)$ , which stabilizes the system model within a given terminal region  $\mathbb{X}_f$ , see [49]. In particular, for a disturbance-free LTI system with convex constraints, say

$$x(k+1) = Ax(k) + B\bar{u}(k), \quad (24)$$

using quadratic MPC, one can use a fixed state feedback gain as a stabilizing controller for the terminal region  $\mathbb{X}_f$  (see Section 2.5 in [49]). To adapt this idea to systems with finite control inputs, we first introduce an associated convex set via:

$$\bar{\mathbb{U}} \triangleq \{\bar{u} \in \mathbb{R}^m : |\bar{u}| \leq \bar{u}_{\max}\},$$

where  $\bar{u}_{\max} \in (0, \infty)$  is a design parameter. Since  $\bar{\mathbb{U}}$  is bounded, so is the quantization effect, i.e.,

$$\Delta_q \triangleq \max_{\bar{u} \in \bar{U}} |q_U(\bar{u}) - \bar{u}| < \infty. \quad (25)$$

Note that  $\Delta_q$  depends upon  $\bar{u}_{\max}$ .

Based on this, stability of FCS-MPC can be examined by investigating properties of a local controller  $\kappa_f(x)$  corresponding to the optimal solution presented in (22) with prediction horizon  $N = 1$ . In this case, one has  $F = B^T PA$  and  $W = B^T PB + R$ , so that

$$\mathbf{u}_{uc}^{\text{opt}}(x) = Kx, \quad K = -W^{-1}F. \quad (26)$$

The above motivates one to impose that the terminal state in the optimisation lies inside a terminal region:  $x(k+N) \in \mathbb{X}_f$ , with

$$\mathbb{X}_f \triangleq \{x \in \mathbb{R}^n : |x| \leq b\}, \quad b \triangleq \frac{\bar{u}_{\max}}{|K|}. \quad (27)$$

Within this region the local controller satisfies

$$\kappa_f(x) = Kx + H^{-1}\eta_V(x), \quad x \in \mathbb{X}_f, \quad (28)$$

where  $\eta_V(x) \triangleq \bar{\eta}_V(W^{-1/2}Kx)$ . Clearly,

$$\begin{aligned} |\eta_V(x)| &\leq |q_V(HKx) - HKx| \leq |Hq_U(Kx) - HKx| \\ &\leq |H||q_U(Kx) - Kx| \leq |H|\Delta_q, \end{aligned} \quad (29)$$

where we have used (25).

Consequently, system (24) with the proposed local controller  $\kappa_f(x)$  in (28) can be expressed via:

$$x(k+1) = A_K x(k) + w_f(x(k)), \quad \forall x(k) \in \mathbb{X}_f, \quad (30)$$

where  $A_K = A + BK$ , and  $w_f(x(k)) = BH^{-1}\eta_V(x(k))$  represents the effect of the quantization on the ‘‘nominal system’’,  $x(k+1) = A_K x(k)$ .

Notice that, in (30),  $w_f(x)$  is not an external disturbance but a known discontinuity produced by the quantization, which makes (30) a nonlinear system. The key point here is that  $w_f(x)$  is bounded on  $\mathbb{X}_f$ . Therefore, the local controller can be shown to be stabilizing if it is robust to bounded input disturbances. As shown in [3], it is convenient to choose the matrix  $P$  in (10) as the (unique) solution to the algebraic Riccati equation

$$A_K^T P A_K + Q + K^T R K - P = 0. \quad (31)$$

With this choice,  $\kappa_f(x)$  in (28) can be used to guarantee closed-loop stability of FCS-MPC. Theorem 2, given below, establishes that for all  $x(0)$  that belong to the feasible set  $X_N$ , the system will be steered by the multi-step predictive controller towards the terminal region  $\mathbb{X}_f \subseteq X_N$  and then (with the same controller) into an ultimately bounded set  $\mathcal{D}_{\delta_N} \subset \mathbb{X}_f$ .

**Theorem 2 ([3])** Let  $\mathcal{D}_{\delta_N} \triangleq \{x \in \mathbb{X}_f : |x| \leq \delta_N\}$  be a neighbourhood of the origin, where

$$\delta_N^2 \triangleq \gamma_N \Delta_q^2, \quad \gamma_N \triangleq \left( \frac{1 + (1 - \rho)N}{\lambda_{\min}(Q)(1 - \rho)} \right) |W|. \quad (32)$$

Suppose that  $x(0) \in X_N$  and the matrix  $P$  in (10) satisfies (31). If  $\Delta_q$  in (25) is bounded by

$$\Delta_q^2 < \frac{b^2}{\gamma_N}, \quad (33)$$

then  $\limsup_{k \rightarrow \infty} |x(k)| \leq \delta_N$ . Furthermore, there exists a finite instant  $t > 0$ , such that after that instant, the system state  $x(k)$  converges at an exponential rate, i.e., there exists  $c > 0$  and  $\rho \in [0, 1)$ , such that

$$|x(k)|^2 \leq c\rho^{k-t}|x(t)|^2 + \gamma_N \Delta_q^2, \quad \forall k \geq t, \quad (34)$$

where  $c = \lambda_{\max}(P)/\lambda_{\min}(Q)$  and  $\rho = 1 - 1/c$ , with  $\lambda_{\min}(Q) \leq \lambda_{\max}(P)$ .

### 4.3 Example: Reference Tracking

The topology of a two-level inverter is presented in Fig. 7. The associated continuous-time dynamic model for the three-phase output current,  $i_{abc} \triangleq [i_a \ i_b \ i_c]^T$ , is

$$\frac{di_{abc}(t)}{dt} = -\frac{r}{L}i_{abc}(t) + \frac{1}{L}(V_{dc}S_{abc}(t) - v_o(t)I_{3 \times 1}), \quad (35)$$

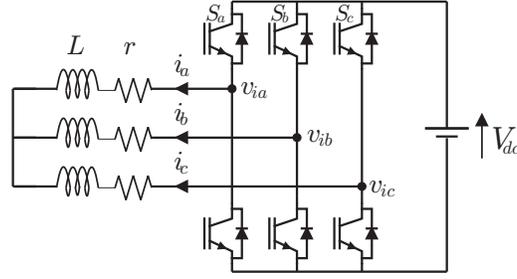
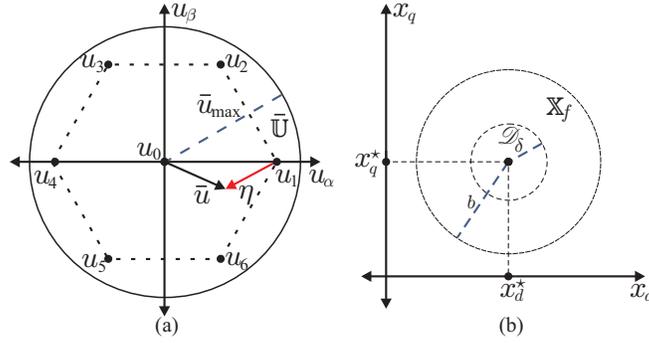
where  $V_{dc}$  denotes the dc-link voltage and  $v_o$  stands for the common-mode voltage. The latter is defined as  $v_o = \frac{1}{3}(v_a + v_b + v_c)$ , where  $v_a$ ,  $v_b$  and  $v_c$  are the voltages at the inverter terminals, see Fig. 7. The switch positions,  $S_{abc} \triangleq [S_a \ S_b \ S_c]^T$ , belong to the following finite set

$$\left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}. \quad (36)$$

For this converter, the control target is to track three-phase sinusoidal references of the form:

$$i_{abc}^*(t) = a^* [\sin(\omega t) \ \sin(\omega t - 2\pi/3) \ \sin(\omega t + 2\pi/3)]^T \quad (37)$$

We will next illustrate how the preceding ideas can be applied to this situation. For that purpose, we first note that sinusoidal quantities in a three-phase system can be transformed into a rotating orthogonal  $dq$  reference frame using the so-called Park transformation. More specifically, the three-phase current  $i_{abc}$  in (35) is transformed into the  $dq$  frame by the transformation


**Fig. 7** Two-level inverter topology.

**Fig. 8** Sets involved in the cost function design; (a) finite control set  $\mathbb{U}$  and nominal input set  $\bar{\mathbb{U}}$ ; (b) terminal region  $\mathbb{X}_f$  and bounded set  $\mathcal{D}_\delta$ .

$$i_{dq}(t) = \Gamma(t)i_{abc}(t), \quad (38)$$

where:

$$\Gamma(t) \triangleq \frac{2}{3} \begin{bmatrix} \sin(\omega t) & \sin(\omega t - \frac{2\pi}{3}) & \sin(\omega t + \frac{2\pi}{3}) \\ \cos(\omega t) & \cos(\omega t - \frac{2\pi}{3}) & \cos(\omega t + \frac{2\pi}{3}) \end{bmatrix}, \quad (39)$$

and  $i_{dq} \triangleq [i_d \ i_q]^T$ .

Fig. 8 (left) shows the typical output voltages of a two-level inverter in a vectorial representation in the stationary orthogonal  $\alpha\beta$  coordinate system. The finite input set,  $\mathbb{U}$ , contains the 7 inverter vectors, which are contained by the nominal input set,  $\bar{\mathbb{U}}$ , i.e.,

$$\mathbb{U} = \{u_0, \dots, u_6\} \subset \bar{\mathbb{U}} \subset \mathbb{R}^2. \quad (40)$$

In this case, the quantization of the nominal input  $\bar{u} \in \bar{\mathbb{U}}$  is given by  $q_{\mathbb{U}}(\bar{u}) = u_1$ , thus  $\eta_{\mathbb{U}}(\bar{u}) = u_1 - \bar{u}$ . Notice that the inverter voltage vectors rotate in the  $dq$  reference frame. However, they always will be contained by the nominal input set,  $\bar{\mathbb{U}}$ , producing the same maximum quantization error  $\Delta_q$  as in the  $\alpha\beta$  coordinate system.

Thus, considering  $x = i_{dq}$  and  $u = s_{dq}$ , the discrete-time model of the two-level inverter in the  $dq$  frame is

$$x(k+1) = Ax(k) + Bu(k), \quad u(k) \in \mathbb{U}(k), \quad (41)$$

$$A = \begin{bmatrix} 1 - hr/L & \omega h \\ -\omega h & 1 - hr/L \end{bmatrix}, \quad B = (h/L)V_{dc}I_{2 \times 2}, \quad (42)$$

where  $h$  is the sampling period and

$$\mathbb{U}(k) = \Gamma(kh)\mathbb{S}. \quad (43)$$

In this case, with a current reference of constant amplitude  $a^*$ , the reference

$$x^* = i_{dq}^* = [a^* \ 0]^T \quad (44)$$

directly follows. The input required to keep this state value is given by

$$u^* = S_{dq}^* x^* = [ra^*/V_{dc} \quad \omega La^*/V_{dc}]^T. \quad (45)$$

Here, experimental results of the performance of FCS-MPC when applied to a three-phase two-level inverter are presented. The inverter prototype was built using discrete insulated-gate bipolar transistors (IGBTs) IRG4PC30KD. The electrical parameters of the converter-load system are  $V_{dc} = 200 \text{ V}$ ,  $r = 5 \ \Omega$  and  $L = 17 \text{ mH}$ , see Fig. 7. The predictive strategy was implemented in a standard TMS320C6713 DSP considering a sampling period of  $h = 100 \ \mu\text{s}$ . The desired amplitude for the output current is  $a^* = 5 \text{ A}$  with an angular frequency of  $\omega = 2\pi 50 \text{ rad/s}$ .

Following the result in Theorem 1, one obtains for the weighting matrices  $Q = I_{2 \times 2}$  and  $R = 2I_{2 \times 2}$  that

$$P = 1.7455I_{2 \times 2}, \quad K = \begin{bmatrix} -0.4514 & -0.0146 \\ 0.0146 & -0.4514 \end{bmatrix}. \quad (46)$$

A key observation is that the time-varying constraint set  $\mathbb{U}$  in (43), can be bounded by a fixed nominal set  $\bar{\mathbb{U}}$ . In Fig. 8, one can see that when the nominal input  $\bar{u}$  is inside the hexagon-shaped boundary, the maximum quantization error,  $\Delta_q$ , is given by the centroid of the equilateral triangle formed by the adjacent inverter vectors. Therefore, the maximum quantization error is given by  $\Delta_q = 2\frac{\sqrt{3}}{9}$ . The associated nominal input set can be chosen as:

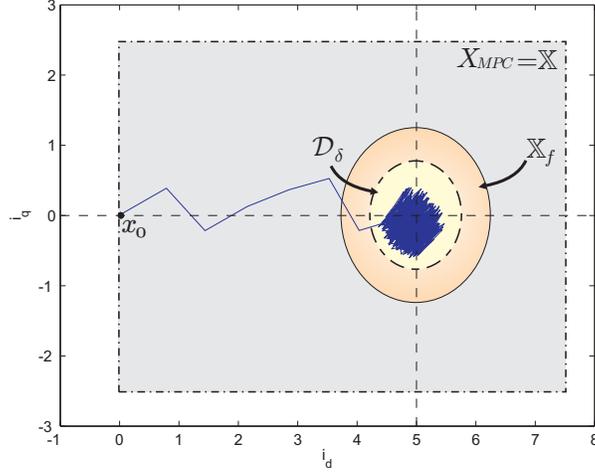
$$\bar{\mathbb{U}} \triangleq \{\bar{u} \in \mathbb{R} : |\bar{u}| \leq 2\Delta_q\},$$

while the terminal region can be characterized via (see [4] for details):

$$\mathbb{X}_f \triangleq \left\{ x \in \mathbb{R}^n : |x - x^*| \leq \frac{u_{\max} - |u^*|}{|K|} = 1.3 \right\}$$

which provides that

$$|\eta_{\mathbb{U}}(\bar{u})| \leq \Delta_q = 2\frac{\sqrt{3}}{9}, \quad \forall x \in \mathbb{X}_f.$$



**Fig. 9** Convergence of the two-level inverter for  $R = 0.0001I_{2 \times 2}$ .

Thus, one can anticipate that the system state will be led by the predictive controller to the ultimately invariant set:

$$\mathcal{D}_{\delta_N} \triangleq \{x \in \mathbb{R}^n : |x - x^*| \leq \delta = 0.8088\}. \quad (47)$$

The evolution of the two-level inverter using FCS-MPC with  $N = 1$  and starting from  $i_d = i_q = 0$  is depicted in Fig. 9. Here, one can see that the predictive controller leads the system state to the terminal region,  $\mathbb{X}_f$ , and then to  $\mathcal{D}_{\delta_N}$ . As expected for this kind of controller, the inverter voltage spectrum is spread, as can be observed in Fig. 10. If this is undesired, then one can use noise shaping techniques, as described in [10, 47].

## 5 An Efficient Algorithm for Finite-Control Set MPC

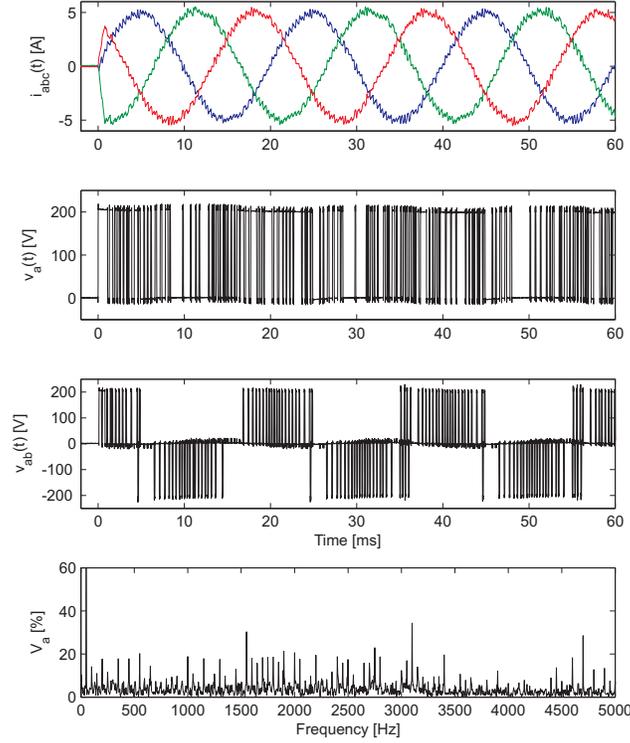
In this section we consider the cost function

$$V(x(k), \mathbf{u}(k)) = \sum_{\ell=k}^{k+N-1} (y^*(\ell+1) - y'(\ell+1))^T (y^*(\ell+1) - y'(\ell+1)) + \lambda_u (\Delta u'(\ell))^T \Delta u'(\ell), \quad (48)$$

which penalises the predicted output errors and the control effort

$$\Delta u'(\ell) \triangleq u'(\ell) - u'(\ell - 1).$$

The latter is weighted by the non-negative scalar weighting factor  $\lambda_u$ . The cost function (48) is minimized subject to



**Fig. 10** System state and input trajectories, and inverter voltage spectrum.

$$\begin{aligned} \mathbf{u}(k) &\in \mathbb{U}^N \\ \|\Delta \mathbf{u}'(\ell)\|_\infty &\leq 1, \quad \forall \ell \in \{k, k+1, \dots, k+N-1\}, \end{aligned} \quad (49)$$

where the first constraint restricts the sequence of manipulated variables to the set of feasible switch positions of the converter. In many converters the second constraint is required to avoid switching in a phase by more than one step up or down.

Owing to the discrete nature of the decision variable  $\mathbf{u}(k)$ , minimizing (48) subject to (49) is difficult, except for short horizons. In fact, as the prediction horizon is enlarged and the number of decision variables is increased, the (worst-case) computational complexity grows exponentially, thus, cannot be bounded by a polynomial, see also [48]. The difficulties associated with minimizing  $V$  become apparent when using exhaustive search. With this method, the set of admissible switching sequences  $\mathbf{u}(k)$  is enumerated and the cost function evaluated for each such sequence. The switching sequence with the smallest cost is (by definition) the optimal one and its first element is chosen as the control input.

It is easy to see that exhaustive search is computationally feasible only for very small horizons  $N$ , such as one or two. In fact, for  $N = 5$ , assuming a three-level converter, the number of switching sequences amounts to  $1.4 \cdot 10^7$ .

Techniques from vector quantization [18] and from mathematical programming, such as branch and bound [19, 34, 44], can be used to reduce the computational burden. However, none of the general methods take advantage of the particular structure of (48) and the fact that in MPC the solution is implemented in a moving horizon manner.

To address computational issues, we will exploit the geometrical structure of the underlying MPC optimization problem and present a practical optimization algorithm. The algorithm uses elements of sphere decoding [28] to provide optimal switching sequences, requiring only little computational resources, thus, enabling the use of longer prediction horizons in practical applications [5, 23, 24].

We will illustrate the ideas on a variable speed drive application consisting of a three-level neutral point clamped voltage source inverter driving an induction machine. The methods proposed and results obtained are directly applicable to both the machine-side inverter in an ac drive setting, as well as to grid-side converters. The ideas can also be used for other converter topology and are particularly promising for topologies with a high number of voltage levels.

### 5.1 Modified Sphere Decoding Algorithm

Using algebraic manipulations akin to those mentioned in Section 4, it is easy to show that the minimization of (48) amounts to finding

$$\mathbf{u}^{\text{opt}}(k) = \arg \min_{\mathbf{u}} (\mathbf{z} - H\mathbf{u})^T (\mathbf{z} - H\mathbf{u}), \quad \text{subject to (49),} \quad (50)$$

where  $H$  is an invertible lower-triangular matrix. In (50), we use

$$\mathbf{z} = H\mathbf{u}^{\text{uc}},$$

where  $\mathbf{u}^{\text{uc}}$  is the sequence obtained from optimizing (48) *without constraints*, i.e., with  $\mathbb{U} = \mathbb{R}^3$ . Thus, we have rewritten the MPC optimization problem as a (truncated) *integer least-squares* problem. Interestingly, various efficient solution algorithms for (50) subject to finite-set constraints have been developed in recent years; see, e.g., [1] and references therein. We will next show how to adapt the sphere decoding algorithm [16, 28] to find the optimal switching sequence  $\mathbf{u}^{\text{opt}}(k)$ .

The basic idea of the algorithm is to iteratively consider candidate sequences, say  $\mathbf{u} \in \mathbb{U}^N$ , which belong to a sphere of radius  $\rho(k) > 0$  centered in  $\mathbf{z}$ ,

$$(\mathbf{z} - H\mathbf{u})^T (\mathbf{z} - H\mathbf{u}) \leq \rho(k). \quad (51)$$

Especially in the case of multilevel converters (where  $\mathbb{U}$  has many elements; see, e.g., [37]), the set of candidate sequences satisfying the above conditions is much smaller than the original constraint set  $\mathbb{U}^N$ . Not surprisingly, computation times can be drastically reduced compared to exhaustive search.

A key property used in sphere decoding is that, since  $H$  is triangular, for a given radius, identifying candidate sequences which satisfy (51) is very simple. In particular, for the present case,  $H$  is lower triangular, thus (51) can be rewritten as

$$\rho^2(k) \geq (z_1 - H_{(1,1)}u_1)^2 + (z_2 - H_{(2,1)}u_1 - H_{(2,2)}u_2)^2 + \dots \quad (52)$$

where  $z_i$  denotes the  $i$ -th element of  $\mathbf{z}$ ,  $u_i$  is the  $i$ -th element of  $\mathbf{u}$ , and  $H_{(i,j)}$  refers to the  $(i, j)$ -th entry of  $H$ . Therefore, the solution set of (51) can be found by proceeding in a sequential manner akin to Gaussian elimination, in the sense that at each step only a one-dimension problem needs to be solved; for details, see [28].

The algorithm requires an initial value for the radius used at time  $k$  to determine  $\mathbf{u}$ . On the one hand, the radius  $\rho(k)$  should be as small as possible, enabling us to remove as many candidate solutions *a priori* as possible. On the other hand,  $\rho(k)$  must not be too small, to ensure that the solution set is non-empty. As shown in [23], it is convenient to choose the initial radius by using the following *educated guess* for the optimal solution:

$$\mathbf{u}^{\text{sub}}(k) = \begin{bmatrix} 0 & I & 0 & \dots & 0 \\ 0 & 0 & I & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & I \\ 0 & \dots & \dots & 0 & I \end{bmatrix} \mathbf{u}^{\text{opt}}(k-1), \quad (53)$$

which is obtained by shifting the previous solution by one time-step and repeating the last switch position. This is in accordance with the moving horizon optimization paradigm. Since the optimal solution at the previous time-step satisfies the constraint,  $\mathbf{u}^{\text{sub}}(k)$  is a feasible solution candidate of (48). Given (53), the initial value of  $\rho(k)$  is then set to:

$$\rho(k) = (\mathbf{z} - H\mathbf{u}^{\text{sub}}(k))^T (\mathbf{z} - H\mathbf{u}^{\text{sub}}(k)). \quad (54)$$

At each time-step  $k$ , the controller first uses the current system state  $\mathbf{x}(k)$ , the future reference values, the previous switch position  $\mathbf{u}(k-1)$  and the previous optimizer  $\mathbf{u}^{\text{opt}}(k-1)$  to calculate  $\mathbf{u}^{\text{sub}}(k)$ ,  $\rho(k)$  and  $\mathbf{z}$ . The optimal switching sequence  $\mathbf{u}^{\text{opt}}(k)$  is then obtained by invoking Algorithm 1 (see [23]):

$$\mathbf{u}^{\text{opt}}(k) = \text{MSPHDEC}(\emptyset, 0, 1, \rho^2(k), \mathbf{z}), \quad (55)$$

where  $\emptyset$  is the empty set<sup>2</sup>.

As can be seen in Algorithm 1, this modification to sphere decoding operates in a recursive manner. Starting with the first component, the switching sequence

<sup>2</sup> The notation  $H_{(i,1:i)}$  refers to the first  $i$  entries of the  $i$ -th row of  $H$ ; similarly,  $\mathbf{u}_{1:i}$  are the first  $i$  elements of the vector  $\mathbf{u}$ . Note that the matrix  $H$  is time-invariant and does not change when running the algorithm. Therefore,  $H$  can be computed once offline before the execution of the algorithm.

**Algorithm 1** Modified sphere decoding algorithm

---

```

function  $\mathbf{u}^{\text{OPT}}(k) = \text{MSPHDEC}(\mathbf{u}, d^2, i, \rho^2, \mathbf{z})$ 
  for each  $u \in \{-1, 0, 1\}$  do
     $u_i \leftarrow u$ 
     $d'^2 \leftarrow (z_i - H_{(i,1:i)} \mathbf{u}_{1:i})^T (z_i - H_{(i,1:i)} \mathbf{u}_{1:i}) + d^2$ 
    if  $d'^2 \leq \rho^2$  then
      if  $i < 3N$  then
         $\text{MSPHDEC}(\mathbf{u}, d'^2, i+1, \rho^2, \mathbf{z})$ 
      else
        if  $\mathbf{u}$  meets (49) then
           $\mathbf{u}^{\text{opt}} \leftarrow \mathbf{u}$ 
           $\rho^2 \leftarrow d'^2$ 
        end if
      end if
    end if
  end for
end function

```

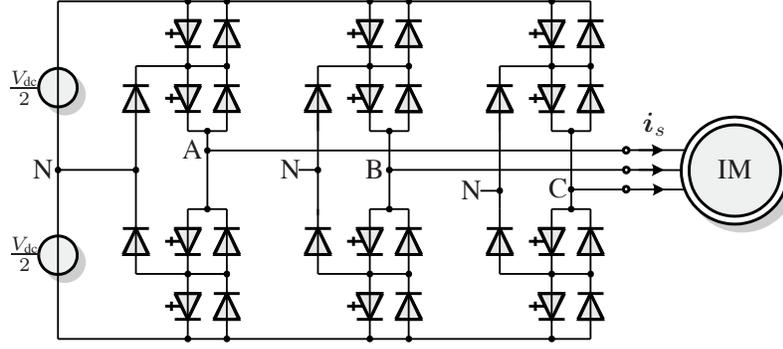
---

$\mathbf{u}$  is built component by component, by considering the admissible single-phase switch positions in the constraint set  $\{-1, 0, 1\}$ . If the associated squared distance is smaller than the current value of  $\rho^2$ , then one proceeds to the next component. If the last component, i.e.,  $u_{3N}$ , has been reached, meaning that  $\mathbf{u}$  is of full dimension  $3N$ , then  $\mathbf{u}$  is a candidate solution. If  $\mathbf{u}$  meets the switching constraint (49) and if the distance is smaller than the current optimum, then one updates the incumbent optimal solution  $\mathbf{u}^{\text{opt}}$  and also the radius  $\rho$ .

The computational advantages of this algorithm stem from adopting the notion of branch and bound [34, 44]. Branching is done over the set of single-phase switch positions  $\{-1, 0, 1\}$ ; bounding is achieved by considering solutions only within the sphere of current radius. If the distance  $d'$  exceeds the radius, a certificate has been found that the branch (and all its associated switching sequences) provides only solutions worse than the incumbent optimum. Therefore, this branch can be pruned, i.e., removed from further consideration without exploring it. During the optimization procedure, whenever a better incumbent solution is found, the radius is reduced and the sphere thus tightened, so that the set of candidate sequences is as small as possible, but non-empty. The majority of the computational burden relates to the computation of  $d'$  via evaluating the terms  $H_{(i,1:i)} \mathbf{u}_{1:i}$ . Thanks to (52),  $d'$  can be computed sequentially, by computing only the squared addition due to the  $i$ th component of  $\mathbf{u}$ . In particular, the sum of squares in  $d$ , accumulated over the layers 1 to  $i-1$ , does not need to be recomputed.

## 5.2 Simulation Study of FCS-MPC

As an illustrative example of a power electronics system, we consider a medium-voltage variable speed drive system consisting of a neutral point clamped (NPC)



**Fig. 11** Three-level three-phase neutral point clamped voltage source inverter driving an induction motor with a fixed neutral point potential

voltage source inverter (VSI) and a squirrel-cage induction machine (IM). This setup is shown in Fig. 11. The inverter can synthesize three output voltage levels at each of its three phase terminals. The total dc-link voltage  $V_{dc}$  is assumed constant and the neutral point potential  $N$  is fixed.

### System Model

Let the integer variables  $u_a, u_b, u_c \in \{-1, 0, 1\}$  denote the switch positions in the three phase legs. The voltage vector applied to the machine terminals in the stationary orthogonal  $\alpha\beta$  coordinate system is

$$v_{s,\alpha\beta} = \begin{bmatrix} v_{s\alpha} \\ v_{s\beta} \end{bmatrix} = \frac{1}{2} V_{dc} \mathcal{P} u \quad (56)$$

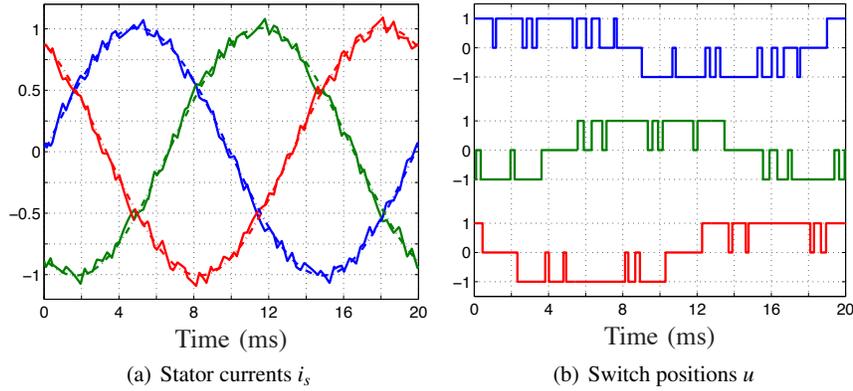
with

$$\mathcal{P} \triangleq \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}, \quad u \triangleq \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} \in \mathbb{U} \quad \text{and} \quad \mathbb{U} \triangleq \{-1, 0, 1\}^3. \quad (57)$$

For the state-space model of an induction machine in the stationary coordinate system, we choose the stator currents  $i_{s\alpha}$  and  $i_{s\beta}$  and the rotor flux linkages  $\psi_{r\alpha}$  and  $\psi_{r\beta}$  as state vector

$$x \triangleq [i_{s\alpha} \ i_{s\beta} \ \psi_{r\alpha} \ \psi_{r\beta}]^T.$$

The model input are the stator voltages  $v_{s\alpha}$  and  $v_{s\beta}$  as defined in (56). The model parameters are the stator and rotor resistances  $R_s$  and  $R_r$ , and the stator, rotor and mutual reactances  $X_{ls}$ ,  $X_{lr}$  and  $X_m$ , respectively. Assuming operation at a constant speed, the angular velocity of the rotor,  $\omega_r$ , is also a parameter. The continuous-time state-space equations of the squirrel-cage induction machine are then (see [32])



**Fig. 12** Simulated waveforms for MPC with horizon  $N = 10$  and the weight  $\lambda_u = 0.103$

$$\frac{di_{s,\alpha\beta}}{dt} = -\frac{1}{\tau_s} i_{s,\alpha\beta} + \left( \frac{1}{\tau_r} - \omega_r \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \right) \frac{X_m}{D} \psi_{r,\alpha\beta} + \frac{X_r}{D} v_{s,\alpha\beta} \quad (58a)$$

$$\frac{d\psi_{r,\alpha\beta}}{dt} = \frac{X_m}{\tau_r} i_{s,\alpha\beta} - \frac{1}{\tau_r} \psi_{r,\alpha\beta} + \omega_r \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \psi_{r,\alpha\beta}, \quad (58b)$$

where we have used

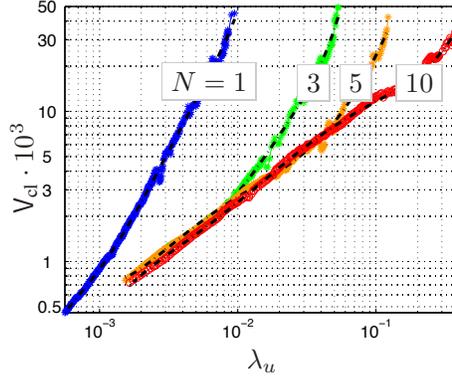
$$X_s \triangleq X_{ls} + X_m, \quad X_r \triangleq X_{lr} + X_m, \quad D \triangleq X_s X_r - X_m^2, \quad \tau_s \triangleq \frac{X_r D}{R_s X_r^2 + R_r X_m^2} \quad \text{and} \quad \tau_r \triangleq \frac{X_r}{R_r}.$$

The objective of the current controller is to manipulate the three-phase switch position  $u$  such that the stator current vector  $i_{s,\alpha\beta}$  closely tracks its reference. To this end, we define the system output vector  $y \triangleq i_{s,\alpha\beta}$  and its reference  $y^* \triangleq i_{s,\alpha\beta}^*$ . The second control objective is to minimize the switching effort, i.e., the switching frequency or the switching losses.

### Performance Evaluation

As an example of a typical medium-voltage induction machine, consider a 3.3 kV and 50 Hz squirrel-cage induction machine rated at 2 MVA with a total leakage inductance of 0.25 pu. The dc-link voltage is  $V_{dc} = 5.2$  kV and assumed to be constant. The parameters of the drive system are provided in [24]. We consider operation at the fundamental frequency 50 Hz and full torque. The controller uses the sampling interval  $h = 25 \mu s$ .

During steady-state operation, the key control performance criteria are the device switching frequency  $f_{sw}$  and the total harmonic distortions (THD) of the current  $I_{THD}$ . We will also investigate the empirical *closed-loop* cost,  $V_{cl}$ , which—in



**Fig. 13** The closed-loop cost is shown as a function of the tuning parameter  $\lambda_u$  for different prediction horizons. The individual simulations are indicated using dots, their overall trend is approximated using dash-dotted polynomials

accordance with (48)—captures the squared RMS current error plus the weighted averaged and squared switching effort.

We start by investigating the steady-state performance of MPC with prediction horizon  $N = 10$  and weighting factor  $\lambda_u = 0.103$ . An average device switching frequency of  $f_{sw} = 300$  Hz results, which is typical for medium-voltage applications, and a current THD of  $I_{THD} = 5.03\%$ . Fig. 12(a) illustrates three-phase stator current waveforms along with their (dash-dotted) references over one fundamental period. The three-phase switch positions are shown in Fig. 12(b).

The influence of  $\lambda_u$  on the empirical closed-loop is investigated next. Steady-state simulations were run for each of the horizons  $N = 1, 3, 5$  and  $10$  and for more than 1000 different values of  $\lambda_u$ , ranging between 0 and 0.5. Focusing on switching frequencies between 100 Hz and 1 kHz, and current THDs below 20%, the results are shown in Fig. 13, using a double logarithmic scale. The cost is significantly reduced as the prediction horizon is increased, suggesting the use of  $N > 1$ .

### Computational Burden

Last, we investigate the computational burden of the modified sphere decoder for different prediction horizons. The switching frequency is held constant at 300 Hz for all prediction horizons by tuning the weight  $\lambda_u$  accordingly. We use the number of switching sequences that are investigated by the algorithm at each time-step as a measure of the computational burden. The average and the maximal number of switching sequences is monitored over multiple fundamental periods. Table 1 shows that the computational burden of the algorithm grows modestly as the prediction horizon is increased, despite being exponential in the worst case. In contrast to that, exhaustive search becomes computationally intractable for prediction horizons exceeding three.

Prediction horizon $N$	Sphere decoding		Exhaustive search	
	avg.	max.	avg.	max.
1	1.18	5	11.8	18
2	1.39	8	171	343
3	1.72	14	2350	4910
5	2.54	35	467'000	970'000
10	8.10	220		

**Table 1** Average and maximal number of switching sequences that need to be considered by the sphere decoding and exhaustive search algorithms to obtain the optimal result, depending on the length of the prediction horizon

## 6 Conclusions

In this chapter, basic aspects and methods underlying model predictive control for power electronics applications have been presented. Algorithms and system theoretic properties depend on whether the discrete switch positions are directly manipulated, or a modulator is used. Special attention has been paid on (practical) stability and computational issues.

Our presentation has been kept at a basic system-theoretic level and was illustrated on simple converter topologies, which can be described via LTI dynamics. Some configurations like, e.g., active front end converters [46] and modular multi-level converters [36], require a more careful consideration of both control theoretic tools and also physical system knowledge for the design of high-performance model predictive controllers.

## References

1. Agrell, E., Eriksson, T., Vardy, A., Zeger, K.: Closest point search in lattices. *IEEE Trans. Inform. Theory* **48**(8), 2201–2214 (2002)
2. Aguilera, R.P., Quevedo, D.E.: Capacitor voltage estimation for predictive control algorithm of flying capacitor converters. In: *IEEE International Conference on Industrial Technology*. Melbourne, Australia (2009)
3. Aguilera, R.P., Quevedo, D.E.: Stability Analysis of Quadratic MPC with a Discrete Input Alphabet. *IEEE Transactions on Automatic Control* (2013)
4. Aguilera, R.P., Quevedo, D.E.: Predictive control of power converters: Designs with guaranteed performance. *IEEE Trans. Ind. Inf.* **11**(1), 53–63 (2015)
5. Baidya, R., Aguilera, R.P., Acuna, P., Vazquez, S., Mouton, H.D.: Multistep model predictive control for cascaded h-bridge inverters: Formulation and analysis. *IEEE Transactions on Power Electronics* **PP**(99), 1–1 (2017). DOI 10.1109/TPEL.2017.2670567
6. Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E.N.: The explicit linear quadratic regulator for constrained systems. *Automatica* **38**(1), 3–20 (2002)
7. Bolognani, S., Bolognani, S., Peretti, L., Zigliotto, M.: Design and implementation of model predictive control for electrical motor drives. *IEEE Trans. Ind. Electron.* **56**(6), 1925–1936 (2009)

8. Carrasco, J.M., Franquelo, L.G., Bialasiewicz, J.T., Galván, E., Guisado, R.C.P., Prats, A.M., León, J.I., Moreno-Alfonso, N.: Power-electronic systems for the grid integration of renewable energy sources: A survey. *IEEE Trans. Ind. Electron.* **53**(4), 1002–1016 (2006)
9. Cortés, P., Kazmierkowski, M.P., Kennel, R.M., Quevedo, D.E., Rodríguez, J.: Predictive Control in Power Electronics and Drives. *IEEE Transactions on Industrial Electronics* **55**(12), 4312–4324 (2008)
10. Cortés, P., Rodríguez, J., Quevedo, D.E., Silva, C.: Predictive current control strategy with imposed load current spectrum. *IEEE Trans. Power Electron.* **23**(2), 612–618 (2008)
11. Cychowski, M., Szabat, K., Orłowska-Kowalska, T.: Constrained model predictive control of the drive system with mechanical elasticity. *IEEE Trans. Ind. Electron.* **56**(6), 1963–1973 (2009)
12. De Doná, J.A., Goodwin, G.C., Serón, M.M.: Anti-windup and model predictive control: Reflections and connections. *European J. Contr.* **6**(5), 467–477 (2000)
13. De Doncker, R.W., Meyer, C., Lenke, R.U., Mura, F.: Power electronics for future utility applications. In: *Proc. IEEE Int. Conf. on Power Electron. and Drive Syst. Bangkok, Thailand* (2007)
14. Emadi, A., Lee, Y.J., Rajashekara, K.: Power electronics and motor drives in electric, hybrid electric, and plug-in hybrid electric vehicles. *IEEE Trans. Ind. Electron.* **55**(6), 2237–2245 (2008)
15. Finch, J.W., Giaouris, D.: Controlled AC electrical drives. *IEEE Trans. Ind. Electron.* **55**(2), 481–491 (2008)
16. Fincke, U., Pohst, M.: Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. *Math. Comput.* **44**(170), 463–471 (1985)
17. Fuentes, E.J., Silva, C.A., Yuz, J.I.: Predictive Speed Control of a Two-Mass System Driven by a Permanent Magnet Synchronous Motor. *IEEE Transactions on Industrial Electronics* **59**(7), 2840–2848 (2012)
18. Gersho, A., Gray, R.M.: *Vector Quantization and Signal Compression*. Kluwer Academic, Boston, MA (1992)
19. Geyer, T.: Computationally efficient model predictive direct torque control. *IEEE Trans. Power Electron.* **26**(10), 2804–2816 (2011)
20. Geyer, T.: *Model predictive control of high power converters and industrial drives*. Wiley, London, UK (2016)
21. Geyer, T., Papafotiou, G., Morari, M.: Model predictive control in power electronics: A hybrid systems approach. In: *Proc. IEEE Conf. Decision Control. Sevilla, Spain* (2005)
22. Geyer, T., Papafotiou, G., Morari, M.: Hybrid model predictive control of the step-down dc-dc converter. *IEEE Trans. Contr. Syst. Technol.* **16**(6), 1112–1124 (2008)
23. Geyer, T., Quevedo, D.E.: Multistep finite control set model predictive control for power electronics. *IEEE Trans. Power Electron.* **29**(12), 6836–6846 (2014)
24. Geyer, T., Quevedo, D.E.: Performance of multistep finite control set model predictive control for power performance. *IEEE Trans. Power Electron.* **30**(3), 1633–1644 (2015)
25. Geyer, T., Torrisi, F., Morari, M.: Optimal complexity reduction of polyhedral piecewise affine systems. *Automatica* **44**(7), 1728–1740 (2008)
26. Goodwin, G.C., Mayne, D.Q., Chen, K., Coates, C., Mirzaeva, G., Quevedo, D.E.: An introduction to the control of switching electronic systems. *Annual Reviews in Control* **34**(2), 209–220 (2010)
27. Grüne, L., Rantzer, A.: On the infinite horizon performance of receding horizon controllers. *IEEE Trans. Automat. Contr.* **53**(9), 2100–2111 (2008)
28. Hassibi, B., Vikalo, H.: On the sphere-decoding algorithm I. Expected complexity. *IEEE Trans. Sign. Process.* **53**(8), 2806–2818 (2005)
29. Holmes, D.G., Lipo, T.A.: *Pulse width modulation for power converters: Principles and practice*. IEEE Press (2003)
30. Holtz, J.: Power electronics—a continuing challenge. *IEEE Ind. Electron. Mag.* **5**(2), 6–15 (2011)

31. Kouro, S., Cortés, P., Vargas, R., Ammann, U., Rodríguez, J.: Model Predictive Control—A Simple and Powerful Method to Control Power Converters. *IEEE Transactions on Industrial Electronics* **56**(6), 1826–1838 (2009)
32. Krause, P.C., Wasynczuk, O., Sudhoff, S.D.: *Analysis of electric machinery and drive systems*, 2nd edn. Wiley (2002)
33. Kvasnica, M., Grieder, P., Baotić, M., Morari, M.: Multi parametric toolbox (MPT). In: R. Alur, G. Pappas (eds.) *Hybrid Syst.: Comput. and Control, LNCS*, vol. 2993, pp. 448–462. Springer, Philadelphia, PA, USA (2004). <http://control.ee.ethz.ch/mpt>
34. Lawler, E.L., Wood, D.E.: Branch and bound methods: A survey. *Op. Res.* **14**(4), 699–719 (1966)
35. Lehman, B., Bass, R.M.: Extensions of averaging theory for power electronic systems. *IEEE Trans. Power Electron.* **11**(4), 542–553 (1996)
36. Lesnicar, A., Marquardt, R.: An innovative modular multilevel converter topology suitable for a wide power range. In: *Proc. IEEE Power Tech. Conf. Bologna, Italy* (2003)
37. Lezana, P., Aguilera, R.P., Quevedo, D.E.: Model predictive control of an asymmetric flying capacitor converter. *IEEE Trans. Ind. Electron.* **56**(6), 1839–1846 (2009)
38. Linder, A., Kennel, R.: Model predictive control for electrical drives. In: *Proc. IEEE Power Electronics Specialists Conference (PESC)*, pp. 1793–1799. Recife, Brazil (2005)
39. Mariéthoz, S., Beccuti, A., Papafotiou, G., Morari, M.: Sensorless explicit model predictive control of the dc-dc buck converter with inductor current limitation. In: *Proc. App. Power Electron. Conf. and Expo.*, pp. 1710–1715 (2008)
40. Mariéthoz, S., Domahidi, A., Morari, M.: High-bandwidth explicit model predictive control of electrical drives. *IEEE Trans. Ind. Appl.* **48**(6), 1980–1992 (2012)
41. Mariéthoz, S., Morari, M.: Explicit model predictive control of a PWM inverter with an *LCL* filter. *IEEE Trans. Ind. Electron.* **56**(2), 389–399 (2009)
42. *Markets and Markets: Power electronics market—global forecast to 2022*. Tech. rep. (2017)
43. Mayne, D.Q., Rawlings, J.B., Rao, C.V., Sokaert, P.O.M.: Constrained model predictive control: Optimality and stability. *Automatica* **36**(6), 789–814 (2000)
44. Mitten, L.G.: Branch-and-bound methods: General formulation and properties. *Op. Res.* **18**(1), 24–34 (1970)
45. Müller, C., Quevedo, D.E., Goodwin, G.C.: How good is quantized model predictive control with horizon one? *IEEE Trans. Automat. Contr.* **56**(11), 2623–2638 (2011)
46. Quevedo, D.E., Aguilera, R.P., Pérez, M.A., Cortés, P., Lizana, R.: Model predictive control of an AFE rectifier with dynamic references. *IEEE Trans. Power Electron.* **27**(7), 3128–3136 (2012)
47. Quevedo, D.E., Goodwin, G.C.: Control of EMI from switch-mode power supplies via multi-step optimization. In: *Proc. Amer. Contr. Conf.*, vol. 1, pp. 390–395. Boston, MA (2004)
48. Quevedo, D.E., Goodwin, G.C., De Doná, J.A.: Finite constraint set receding horizon quadratic control. *Int. J. Robust Nonlin. Contr.* **14**(4), 355–377 (2004)
49. Rawlings, J., Mayne, D.: *Model Predictive Control: Theory and Design*. Nob Hill Publishing (2009)
50. Rodríguez, J., Cortés, P.: *Predictive Control of Power Converters and Electrical Drives*, 1 edn. Wiley-IEEE Press (2012)
51. Rodríguez, J., Kazmierkowski, M.P., Espinoza, J., Zanchetta, P., Abu-Rub, H., Young, H.A., Rojas, C.A.: State of the Art of Finite Control Set Model Predictive Control in Power Electronics. *IEEE Transactions on Industrial Informatics* **9**(2), 1003–1016 (2013)
52. Rodríguez, J., Pontt, J., Silva, C., Correa, P., Lezana, P., Cortés, P., Ammann, U.: Predictive Current Control of a Voltage Source Inverter. *IEEE Transactions on Industrial Electronics* **54**(1), 495–503 (2007)
53. Senesky, M., Eirea, G., Koo, T.J.: Hybrid modelling and control of power electronics. In: A. Pnueli, O. Maler (eds.) *Hybrid Syst.: Comput. and Control, LNCS*, vol. 2623, pp. 450–465. Springer (2003)
54. Vazquez, S., Rodríguez, J., Rivera, M., Franquelo, L.G., Norambuena, M.: Model Predictive Control for Power Converters and Drives: Advances and Trends. *Industrial Electronics, IEEE Transactions on* **PP**(99), 1–1 (2016)