# Direct Model Predictive Control for Power Electronics with Long Prediction Horizons

Petros Karamanakos, *Member, IEEE*, Tobias Geyer, *Senior Member, IEEE*, Nikolaos Oikonomou, *Member, IEEE*, Frederick D. Kieferndorf, *Member, IEEE* and Stefanos Manias, *Fellow, IEEE*

**Abstract**

Direct model predictive control (MPC) strategies that achieve long prediction horizons with a modest computational complexity are reviewed in this article, focusing on power electronics applications. In many MPC problems, a long prediction horizon is required to ensure an adequate closed-loop performance in steady-state and to avoid stability issues. However, the computational effort of solving the optimization problem underlying MPC problems with long prediction horizons is often very large, making the implementation of such schemes in real-time a difficult and challenging task. To overcome this difficulty, three established methodologies are surveyed that yield long prediction horizons with a modest computational burden. Case studies are investigated to substantiate the merits of these schemes. More specifically, for dc-dc boost converters, a move blocking strategy is reviewed, and for ac medium voltage (MV) drives, both an extrapolation and an event-based horizon strategy are examined.

## I. Introduction

In part, due to their switching nature, power electronic systems constitute nonlinear systems with multiple-inputs and multiple-outputs (MIMO), and subject to constraints (e.g. the duty cycle should be limited between zero and one). Over the years many control strategies have been proposed that are mainly based on conventional proportional-integral-derivative (PID) controllers combined with nonlinear techniques, such as pulse width modulation (PWM). However, since PID controllers are best suited to linear, single-input, single-output (SISO), unconstrained control problems, the design procedure of PID controllers for nonlinear, MIMO plants with constraints

P. Karamanakos is with the Institute for Electrical Drive Systems and Power Electronics, Technische Universität München, 80333 Munich, Germany (e-mail: p.karamanakos@ieee.org).

T. Geyer, N. Oikonomou, and F. D. Kieferndorf are with ABB Corporate Research, 5405 Baden-Dättwil, Switzerland (e-mails: t.geyer@ieee.org; nikolaos.oikonomou@ch.abb.com; frederick.kieferndorf@ch.abb.com)

S. Manias is with the Department of Electrical and Computer Engineering, National Technical University of Athens, 15780 Zografou, Athens, Greece (e-mail: manias@central.ntua.gr)

becomes cumbersome [1], [2]. Moreover, controllers of this type are usually tuned to achieve satisfactory performance only in a narrow operating range; outside this range the performance significantly deteriorates. Therefore, the problems associated with many power electronics applications and their closed-loop performance still poses theoretical and practical challenges.

A control algorithm that has recently been gaining popularity in the field of power electronics is model predictive control (MPC) [3], [4]. MPC is a control strategy that was developed in the 1970s in the process control industry as an alternative strategy to conventional PID control. Its success is based on the fact that it uses a mathematical model of the plant, which allows the controller to predict the impact of its control actions [5]. Furthermore, MPC is capable of handling complex and nonlinear dynamics, and constraints can be explicitly included in a simple and effective manner [6]. By imposing constraints on the variables of concern the plant is able to operate very near its physical limits without violating them. Hence, the most favorable operation can be obtained with minimum oversizing of the system, while the operational limits of the plant are fully respected[1]. Thus, MPC has attracted the interest and attention of research and academic communities. Furthermore, the continued advance of powerful microprocessors with increasing computational capabilities has enabled the application of MPC in the field of power electronics with significant success [7]–[11].

In MPC, the designer defines an objective function that captures the control objectives. Moreover, constraints can be imposed on the state variables and/or the manipulated variables, i.e. the control inputs. The underlying optimization problem is solved in real-time at each time-step to determine a plan of control actions over a finite prediction horizon. The sequence of control inputs that minimizes the objective function is the *optimal* solution. Out of this sequence only the first input is applied to the plant. At the next time-step the planning process is repeated with updated measurements or estimates, while the time horizon is shifted one step forward. With this procedure, known as *receding horizon policy* [12], [13], the control strategy introduces feedback.

Generally speaking, in MPC a longer prediction horizon improves stability and plant performance; the longer the prediction interval, the better the control performance is [5][2]. However, it holds true that the computational complexity—depending on the type of the optimization

[1]The variables of interest can be upper, or lower bounded. The constraints that cannot be violated under any circumstances are called *hard constraints*; those that can be violated, but effort should be put into avoiding such violations are called *soft constraints*.

[2]Note that in the field of power electronics the models are typically quite accurate, at least when compared to other engineering disciplines such as the process control. As a result, when using long prediction intervals, the accuracy of the predictions is high. Moreover, the receding horizon policy in MPC adds feedback and provides MPC with a high degree of robustness.

problem—grows exponentially with the length of the prediction horizon and the number of the manipulated variables [14]. Since, the optimization problem must be solved at each time-step, the time required to solve demanding MPC problems is often much longer than the usual sampling interval used in power electronics applications (in the order of microseconds). Therefore, in order to facilitate the real-time implementation of MPC-based algorithms, strategies need to be employed that effectively balance the trade-off between the length of the prediction horizon and the number of required computations.

The necessity for utilizing such strategies in the field of power electronics comes also from the fact that MPC is very often implemented as a direct control method, i.e. the switches of the converters are directly manipulated without the presence of a modulator. Despite the fact that direct MPC strategies require higher computational burden[3] compared to modulator-based MPC schemes, such as generalized predictive control (GPC) [15], [16], these strategies are preferred since the implementation procedure is more straightforward and simpler. In direct MPC the optimization problem is solved in most cases using an enumeration strategy [7]–[9], [11][4]. All possible combinations of the discrete-valued control inputs over the prediction horizon $N$ are enumerated, yielding the so-called *switching sequences*. Subsequently, for each sequence the evolution of the variables of concern is calculated using the prediction model of the plant and the formulated objective function is evaluated. Then, the switching sequence with the minimum associated cost is chosen as the optimal.

In this review article[5], computationally efficient methods that achieve long prediction intervals applicable to direct MPC algorithms for power electronics applications are discussed. Three strategies are highlighted, namely the move blocking strategy, the extrapolation strategy, and the event-based horizon strategy. The selected examples are chosen to show the effectiveness of these established strategies, as well as to indicate the variety of problems that can be addressed. In Section II the move blocking strategy is demonstrated with dc-dc boost converters, while the extrapolation strategy and the event-based horizon strategy are applied to ac medium-voltage (MV) drives in Sections III and IV, respectively.

[3]This statement holds true when a long prediction horizon is used, i.e. for cases where $N > 1$.

[4]An alternative to enumeration strategy is to use precalculated switching sequences, as recently proposed in [17]. The open-loop switching sequences are modified in real-time by shifting the switching instants in order to meet the control objectives, by adopting the principles of constrained optimal control and receding horizon policy.

[5]A preliminary version can be found in [18].

## II. Move Blocking Strategy

As mentioned earlier, in many cases to successfully tackle a control problem in the field of power electronics a long prediction interval $NT_s$ is required, where $N$ is the number of prediction steps, and $T_s$ the sampling interval. However, simply increasing the number of prediction steps $N$ results in an exponential increase of the calculations required. On the other hand, one could increase the sampling interval $T_s$, while keeping $N$ low, but this leads to unacceptable resolution of the possible switching instants, since switching can only occur at the sampling instants. Thus, neither option is viable. One technique to overcome this apparent contradiction and to achieve a long prediction horizon with fewer time-steps $N$ and a small effective sampling interval $T_s$ is the so-called *move blocking* strategy [19].

The key idea behind the move blocking strategy is to divide the prediction horizon into two parts. The first part of the horizon is $N_1$-steps long, while the second part consists of $N_2$ steps, with $N_1, N_2 \in \mathbb{N}^+$. Consequently, the total number of steps of the prediction horizon is $N = N_1 + N_2$. The first $N_1$ steps of the horizon are sampled with a small sampling interval $T_s$, and in order to achieve a long prediction interval the $N_2$ steps further in the future are sampled more coarsely with a multiple of $T_s$, i.e. with $T_s' = n_s T_s$, where $n_s \in \mathbb{N}^+$. Hence, the total length of the prediction horizon is $N_1 T_s + N_2 T_s' = (N_1 + N_2 n_s) T_s$. It should be pointed out that high resolution sampling of the plant is mainly required near the present time-step, i.e. only for the first $N_1$ steps, since, as already mentioned, the state of the switching devices can only change at the sampling instants. On the other hand, by sampling the model more coarsely for steps farther in the future both the two desired goals can be accomplished:

- An extension of the prediction interval is realized without a subsequent increase in the computational burden.
- A sufficiently long, prediction of the future behavior of the plant is achieved, without decreasing the "effective" timing resolution needed for accurate switching actions. This approximate view suffices, since it will be later refined thanks to the receding horizon policy.

To better understand how the moving block scheme is implemented consider the following case. Assume that the plant under investigation is a hybrid (e.g. continuous-valued state variables, discrete-valued input variables), nonlinear system, the discrete-time state-space model of which

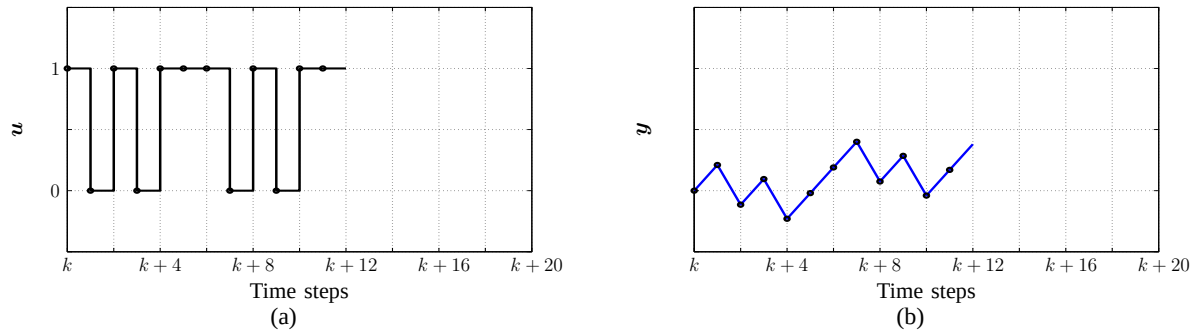



Fig. 1: Without move blocking scheme: The prediction horizon has $N = 12$ time-steps, and the prediction length is $12T_s$.

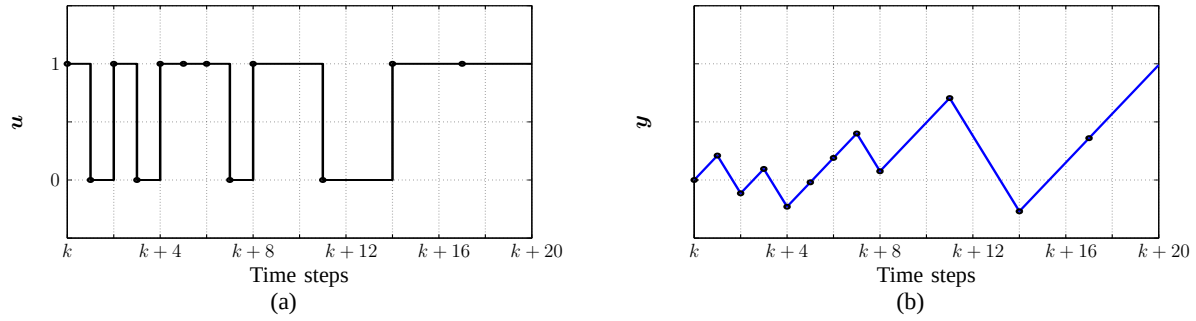



Fig. 2: With move blocking scheme: The prediction horizon has $N = 12$ time-steps, and the prediction length is $20T_s$.

is given by

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

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k)\,, \tag{1b}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{w} \in \mathbb{R}^m$ and $\boldsymbol{y} \in \mathbb{R}$ are the state, disturbance, and output vectors, respectively. The input vector for reasons of simplicity is assumed to be $\boldsymbol{u} \in \{0, 1\}$. Finally, $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is the state matrix, $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ is the disturbance matrix, and the output (row) matrix consists of $n$ elements and it is assumed to be of the form $\boldsymbol{C} = [1\ 0\ \ldots\ 0]$. By setting the input variable $\boldsymbol{u}$ equal to each of its values, two different affine (linear plus offset) expressions are derived:

$$\boldsymbol{x}(k+1) = \begin{cases} \boldsymbol{A}_1\boldsymbol{x}(k) + \boldsymbol{B}_1\boldsymbol{w}(k) & \boldsymbol{u} = 0, \quad \text{Mode “1”} \\ \boldsymbol{A}_2\boldsymbol{x}(k) + \boldsymbol{B}_2\boldsymbol{w}(k) & \boldsymbol{u} = 1, \quad \text{Mode “2”} \end{cases} \tag{2}$$

If we assume a 12-element switching sequence of the form $\boldsymbol{U} = [1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 1]$, and directly apply a single sampling interval for the prediction, the output trajectory $\boldsymbol{Y}$ would be as shown in Fig. 1. On the other hand, if a move blocking scheme is implemented the output trajectory resulting from the same sequence of control moves would be as shown in Fig. 2. Note

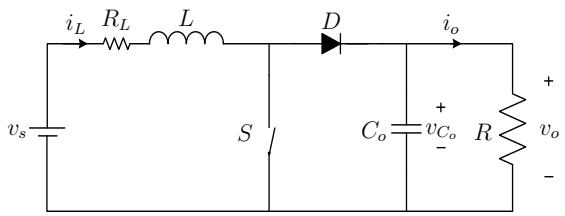



Fig. 3: Topology of the dc-dc boost converter: $v_s$ is the input voltage, and $v_o$ is the output voltage over the load resistor $R$, which is considered equal to the voltage $v_{C_o}$ across the capacitor $C_o$. Furthermore, $i_L$ is the current through the inductor $L$, $R_L$ is its internal resistance, and $i_o$ is the load current. Finally, $S$ and $D$ are the two power switches: $S$ is the controllable one, and $D$ (diode) the uncontrollable.

that for this case $N_1 = 8$, $N_2 = 4$, $n_s = 3$, thus despite the fact that the prediction horizon still has only $N = 12$ time-steps, now the total prediction length is $20T_s$.

Several successful examples of investigations employing the move blocking strategy in power electronics applications can be found in the literature, see [7], [20], [21]. In this survey paper, a dc-dc boost converter example is presented to highlight the efficacy of the move blocking strategy. With this electronic circuit (Fig. 3) the input energy is temporarily stored and then released to the load $R$, by appropriately manipulating the controllable switch $S$. Hence, the converter can produce an output with larger magnitude compared to the (usually unregulated) input dc voltage.

The main control objective is to achieve output voltage regulation despite changes in the input voltage or the load. The standard approach is to indirectly control the output voltage with the inductor current, i.e. an intermediate current loop is added, giving rise to a cascaded control concept [22]. This is due to the fact that the output voltage—in contrast to the input current—exhibits a non-minimum phase behavior with respect to the control input, i.e. the duty cycle[6]. For example, when the output voltage reference is increased, the duty cycle must also increase, but initially the output voltage drops before it begins to rise again.

However, when a voltage-mode controller is designed, i.e. the output voltage is directly controlled by employing only one loop, a sufficiently long prediction horizon is required. In this way, the controller will be able to "see" beyond the initial voltage drop after the step change in the reference and potential closed-loop stability issues can be avoided. Since increasing the prediction horizon will lead to an exponential increase of the computations required, employing a move blocking strategy is an option to reduce this effect [23].

[6]This means that—assuming a linearized model—the control-to-output voltage transfer function contains a right half-plane zero, resulting in a reverse-response system behavior during transients.

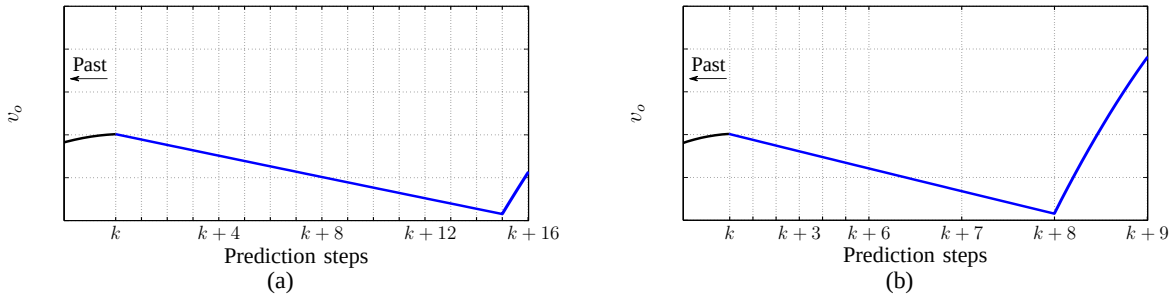



Fig. 4: Effect of the move blocking scheme for the case of a dc-dc boost converter when a step-up change in the output reference voltage occurs at instant $kT_s$. In (a), without move blocking, a prediction horizon of $N = 16$ steps of equal time-intervals is needed. In (b), with the move blocking strategy a $N = 9$ prediction horizon is required ($N_1 = 6$, $N_2 = 3$, and $n_s = 4$, total length $18T_s$).

To demonstrate the effectiveness of the move blocking scheme for the examined case, an illustrative example is shown in Fig. 4. As already explained, when the output voltage reference is stepped up the output voltage initially drops before increasing, due to the non-minimum phase nature of the system. This phenomenon can be clearly seen in Fig. 4. When the output reference voltage is stepped up at time-step $k$ the output voltage should follow this change. However, in order to keep the system stable, and at the same time sample the model with a sufficient resolution, a $16$ time-step horizon is required, i.e. $N = 16$ (see Fig. 4(a)). Assuming that the enumeration strategy[7] is used to solve the optimization problem, the controller must evaluate $2^N$ sequences at every sampling instant [23]. This means, for a $16$-step horizon, the number of the switching sequences to be examined in real-time is $2^{16} = 65536$ and the evolution of the state will be calculated for $16$ steps into the future. On the other hand, with the move blocking scheme a much smaller number of steps is required to achieve the same closed-loop result, see Fig. 4(b). By setting $N_1 = 6$, $N_2 = 3$, and $n_s = 4$ a $9$-step horizon results, the total length of which is $18T_s$. The number of sequences now requiring evaluation is $2^9 = 512$, and the state evolution must only be predicted for $9$ steps. Thus, the computations required are reduced by three orders of magnitude.

To highlight the performance of the closed-loop system the transient behavior during step changes in the output reference voltage is examined. The converter parameters are $L = 450\,\mu$H, $R_L = 0.3\,\Omega$, $C_o = 220\,\mu$F, and $R = 73\,\Omega$. Furthermore, the nominal input voltage is $v_s = 10\,$V. A six-step prediction horizon is implemented, i.e. $N = 6$, and the sampling interval is $T_s = 10\,\mu$s. The prediction horizon is split into $N_1 = 4$ and $N_2 = 2$ with $n_s = 2$.

[7]In Section I the enumeration strategy is briefly presented.

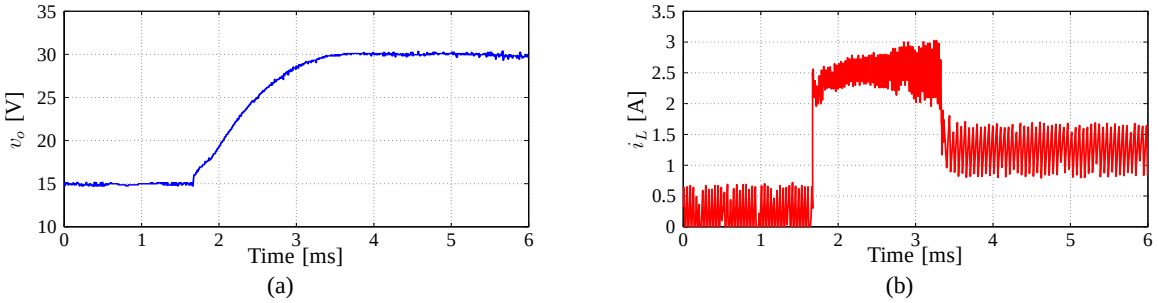



Fig. 5: Closed-loop performance during a step-up change in the output voltage reference: a) output voltage, and b) inductor current (experimental results).

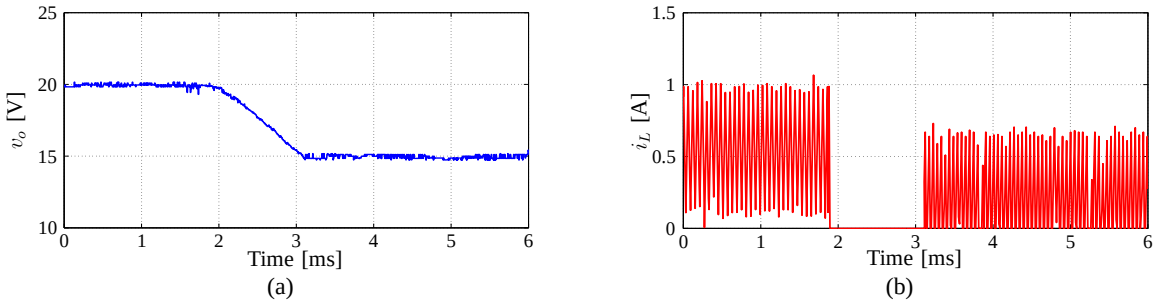



Fig. 6: Closed-loop performance during a step-down change in the output voltage reference: a) output voltage, and b) inductor current (experimental results).

Both, a step-up and a step-down change in the output reference voltage are examined, as shown in Figs. 5 and 6, respectively. For the first case, the reference voltage is stepped up from $v_{o,\text{ref}} = 15\,\text{V}$ to $v_{o,\text{ref}} = 30\,\text{V}$ at time $t \approx 1.7\,\text{ms}$. As is shown in Fig. 5, the current is instantaneously increased to rapidly charge the output capacitor to the new demanded level. The output voltage reaches its new reference value in about $t \approx 1.9\,\text{ms}$ with no observable overshoot. After the transient, the inductor current reduces to its new nominal value corresponding to the steady-state power balance. It should be pointed out, that the controller exhibits this favorable performance thanks to the implemented long prediction horizon; stability is ensured despite the non-minimum nature of the converter.

In the second case, the output reference voltage changes from $v_{o,\text{ref}} = 20\,\text{V}$ to $v_{o,\text{ref}} = 15\,\text{V}$ at $t \approx 1.9\,\text{ms}$. Implementing the MPC algorithm as a direct voltage controller results in the voltage decreasing to its new demanded level as fast as possible. Physically to achieve this, the capacitor should discharge through the load with no input current until it reaches the new reference value. Consequently, the controllable switch is turned off and the inductor current quickly drops to zero. As can be observed, the converter settles to the new operating point in about $t \approx 1.2\,\text{ms}$.

## III. EXTRAPOLATION STRATEGY

A second strategy that can be used to emulate a long prediction horizon, while keeping the computational complexity modest, is *extrapolation* [24]–[29]. Note, that in order to implement this technique soft constraints on the controlled variables, implemented as hysteresis bounds, should be present.

To realize the extrapolation strategy the following types of horizons are defined:

- *Switching horizon*: The switching horizon $N_s \in \mathbb{N}^+$ is the number of steps within which the converter switches can change. It is a constant number and it is set by the designer. The evolution of the variables of concern is calculated over this short horizon for all control input sequences, creating trajectories. Thus, the controlled variables are calculated from time-step $k+1$ to $k+N_s$ based on the sequence of control inputs: $\boldsymbol{U}(k) = [\boldsymbol{u}(k)\ \boldsymbol{u}(k+1) \dots \boldsymbol{u}(k+N_s-1)]$.
- *Prediction horizon*: The prediction horizon $N_p \in \mathbb{N}^+$ includes the switching horizon, i.e. $N_p \geq N_s$, and is of variable length. In order to calculate the length of $N_p$ the most "promising" trajectories [24] are extrapolated from steps $k+N_s-1$ and $k+N_s$, while from step $k+N_s+1$ to $k+N_p-1$ it is assumed that the state of the switches stays the same. Hence, the total length depends on the final slope of each of the extrapolated trajectories; the upper limit is the time-step where the first controlled variable hits a bound. Thus, within this horizon the evolution of the controlled variables is finely calculated within the switching horizon $N_s$ and more coarsely calculated over the extrapolated segment.

To visualize the concepts of the switching and the prediction horizons let us consider again the system given by (1), the simplified model of which is described by (2), by setting either $\boldsymbol{u} = 1$, or $\boldsymbol{u} = 0$. Assume that the control objective is to keep the output variable $\boldsymbol{y}$ within given bounds; $y_{\max}$ is the upper bound, and $y_{\min}$ is the lower bound. In Fig. 7 three different candidate switching sequences are illustrated, each of which result in a different prediction horizon; for the first sequence $N_p = 11$, for the second $N_p = 6$, and for the third $N_p = 14$. However, for all three sequences the switching horizon is the same, i.e. $N_s = 3$.

Direct MPC algorithms employing the extrapolation strategy have been used in the field of power electronics [7], [24], [29] and have been successfully implemented in practice, see e.g. [25]. In this paper, as a case study, consider the five-level active neutral point clamped (ANPC-5L) inverter shown in Fig. 8 driving an induction machine (IM). The ANPC-5L inverter is capable of producing five phase to neutral voltage levels normalized as, $\{-2, -1, 0, +1, +2\}$,

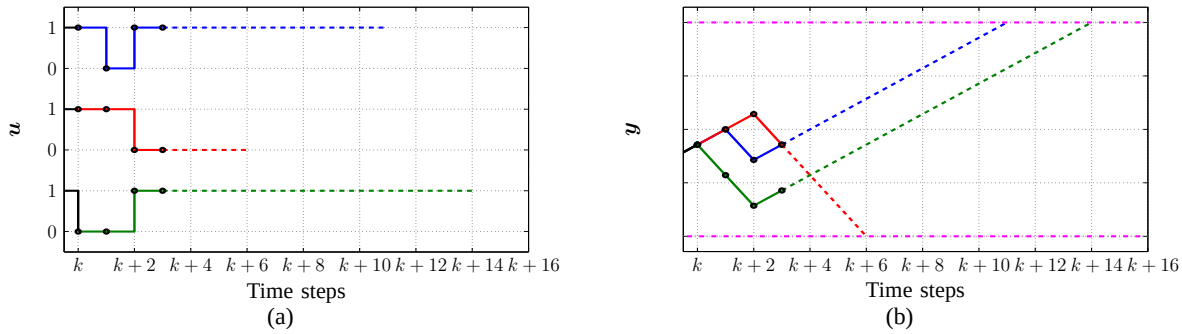



Fig. 7: Three candidate switching sequences and the resulting output trajectories when extrapolation is used. The switching horizon is $N_s = 3$; the trajectories are shown as solid lines within this horizon. The length of the prediction horizon varies, depending on the slope of each of the extrapolated trajectories and on $\boldsymbol{y}(k + N_s)$. For the trajectory shown in blue $N_p = 11$, for the trajectory shown in red $N_p = 6$, and for the green one $N_p = 14$. Finally, the upper $y_{\max}$ and lower $y_{\min}$ bounds are shown with a magenta dash-dotted line.

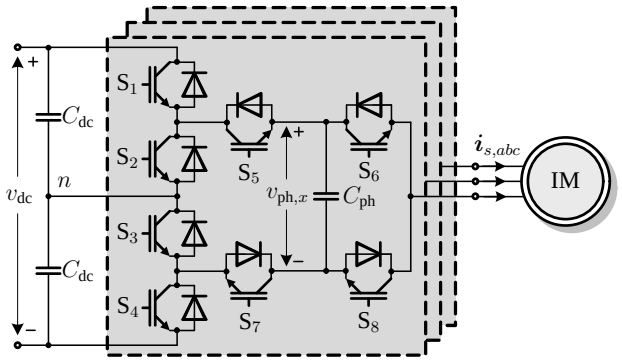



Fig. 8: Circuit diagram of the five-level active neutral point clamped (ANPC-5L) voltage source inverter driving an induction machine (IM).

resulting in $5^3 = 125$ possible three-phase voltage vectors. However, the number of unique line-to-line output voltage vectors is $61$, which means that different three-phase voltage vectors produce the same output voltage. A set with this characteristic is called a three-phase redundancy [30]. Furthermore, the phase leg of the ANPC-5L inverter has eight allowed switching states that can produce the five unique phase to neutral voltage levels [30], i.e. $8^3 = 512$ three-phase vectors that can be produced. Therefore, so called single-phase redundancies exist at the level of the inverter phase leg, as well.

The control objective is to keep the neutral point $v_n$ and phase capacitor voltages $v_{\text{ph},x}$, with $x = \{a, b, c\}$ (see Fig. 8), inside given bounds, while operating the inverter at the lowest possible switching frequency over the whole operating regime. Two types of hysteresis bounds are used, and they are implemented as soft constraints: the inner bounds ($IB$), which are defined by the desired maximum absolute deviation from the respective reference voltage values, and the outer

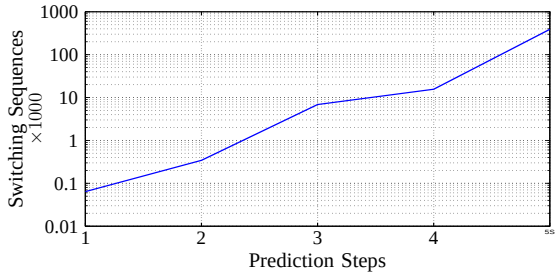



Fig. 9: Number of switching sequences to be examined within a $N_p$-step prediction horizon, with $N_p = \{1, 2, 3, 4\}$, for the case of a ANPC-5L inverter, when the switching constraints are taken into account.

bounds ($OB$), which are set by the allowed physical limits of the semiconductor devices. In order to fulfill the aforementioned goals, the topology single- and three-phase redundancies, i.e. the $512$ generated three-phase voltage vectors, must be effectively exploited.

In order to find the optimal sequence of control actions that meets the control objectives, i.e. the sequence that results in the "best" performance of the plant, all possible switching transitions from one voltage vector to another are enumerated. When an exhaustive search of all candidate sequences is considered, without using the extrapolation strategy, $512^{N_p}$ sequences must be evaluated, where $N_p = N_s$. The hard constraints that stem from the topology of the inverter, such as minimum pulse width duration, IGBT clamp restrictions, and allowed state transitions of the inverter phase leg, can be taken *a priori* into account, and this significantly reduces the allowable (or *feasible*) sequences. The feasible sequences to be considered are generated based on the algorithm presented in [29]. In Fig. 9 the number of sequences to be examined in a $N_p$-step prediction horizon, with $N_p = \{1, 2, 3, 4\}$, is shown. As can be seen, even with a relatively short horizon, such as a three-step horizon ($N_p = 3$), the generated feasible sequences to be evaluated in real-time are too many (for the three-step horizon there are $6859$ sequences). Hence, under these conditions, the implementation of an MPC algorithm in a real-time system seems to be an unattainable task.

However, by employing a two-step switching horizon ($N_s = 2$), and then implementing the extrapolation strategy, there are only $343$ feasible sequences, while at the same time the prediction interval is significantly increased by approximating a long prediction horizon. In Fig. 10, example trajectories are illustrated that can represent either the neutral point or a phase capacitor voltage error, $v_{\text{n,err}} = v_{\text{n,ref}} - v_n$ and $v_{\text{ph},x,\text{err}} = v_{\text{ph},x,\text{ref}} - v_{\text{ph},x}$, respectively. The number of steps in the prediction horizon is determined for each controlled variable by the instant that its extrapolated trajectory crosses one of the inner bounds. The optimal trajectory is the one that corresponds

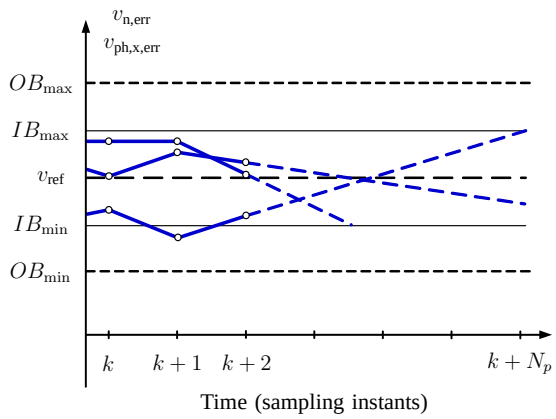



Fig. 10: Examples of internal voltage switching trajectories that illustrate the effect of extrapolation. The switching horizon is $N_s = 2$. The internal voltages are extended by linearly extrapolating the predicted voltage values from steps $k+1$ and $k+2$. Furthermore, the inner ($IB$) and the outer bounds ($OB$) are depicted.

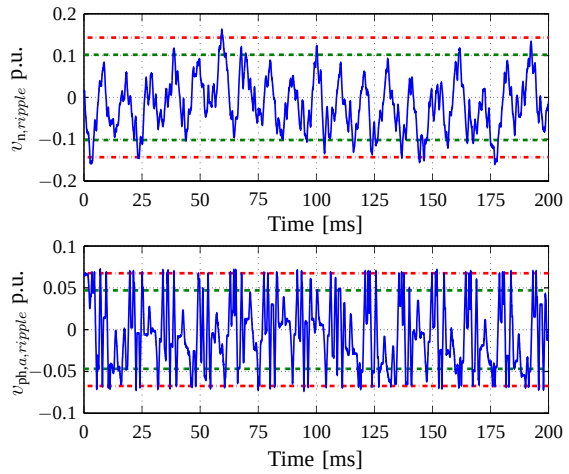



Fig. 11: Simulation results for the per unit ripple of internal voltages with the inner (green dashed line) and outer (red dash-dotted line) bounds. Operating point: $65\%$ speed ($32$ Hz), $42\%$ load.

to the sequence of control inputs that minimizes a specified objective function. In [29], some further approaches to this control problem are presented, as well as different objective functions formulated according to several selection criteria and priority levels. For example, according to [29], the most desirable trajectories are those that lie inside the inner bounds at all time-steps of the prediction horizon. The optimal trajectory is selected from this set of trajectories such that the minimum number of switching transitions occurs. If the set of the most desirable trajectories is empty, then more relaxed selection criteria are defined, and new optimization problems are formulated.

To evaluate the performance of the closed-loop system, the MPC algorithm presented in [29], is simulated using a $1$ MVA ACS 2000 MV drive from ABB coupled to a $6$-kV, $137$-A IM driving

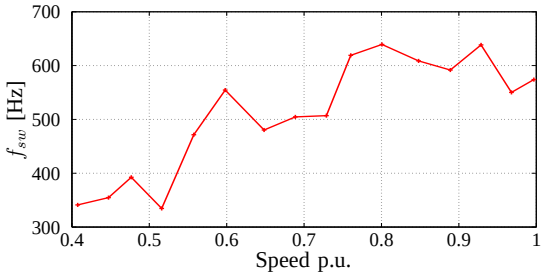



Fig. 12: Simulation results of the switching frequency ($f_{sw}$) over a range of operating points comparing the MPC strategy with the extrapolation strategy as proposed in [29] (red solid line), to the existing algorithm proposed in [30] (blue dashed line).

a quadratic torque load. A two-step switching horizon ($N_s = 2$) is used, with linear extrapolation following the two switching events. The algorithm is executed every $T_s = 25\,\mu$s. In Fig. 11 the waveforms of the voltages of the neutral point and the phase capacitor of phase $a$ are presented. In addition, the switching frequency is kept low over a wide range of operating points, as shown in Fig. 12. For comparison, the switching frequency resulting from the current algorithm used by ABB for this product[8] over the same range of operating points, is also shown in Fig. 12.

Before closing this section, it should be pointed out that even longer prediction horizons can be achieved with the extrapolation strategy. These prediction horizons are of variable length (in the range of $50$ to more than $100$ time-steps) and can be effectively achieved by combining multiple switching horizons (i.e. groups of switch transitions) and extrapolation segments. Therefore, a prediction horizon $N_p$ may consist of multiple "S" and "E" elements, where "S" stands for "switch" and "E" for "extrapolate". Furthermore, there can be an optional extrapolation prior to the first switching event, this is denoted with "e". Thus, with different combinations of "S" and "E" elements, the length of the resulting prediction horizon changes[9], resulting in significantly improved control results, with a minimum increase in the computational burden, as described in [26].

[8]According to the existing strategy, hysteresis bounds are used to limit the neutral point and phase capacitors voltage error without introducing excessive switching. Thus, when the neutral point voltage error is within its inner bounds, no action is taken; the commanded voltage vector is forwarded to the modulator. On the other hand, if it crosses the inner bounds, a vector is selected to balance the neutral point without an additional switching. If the neutral point voltage error crosses its outer bounds a switching occurs in order to keep the voltage in bounds. With regards to the phase capacitors voltages, the single-phase redundancies are considered to keep the error in bounds; the vector that minimizes the error is chosen. However, if there is a conflict with that required for the neutral point voltage balancing then a prioritization is done. For more details the reader is referred to [30].

[9]For example consider a prediction horizon of the type "SSESE". This means that a switching will occur at steps $k$ and $k+1$. Subsequently, the trajectories of all variables will be linearly extrapolated from steps $k+1$ and $k+2$, until one of these hits the bounds. Assume that this happens at step $k+\ell$. At that point another switching will occur followed by a final extrapolation.

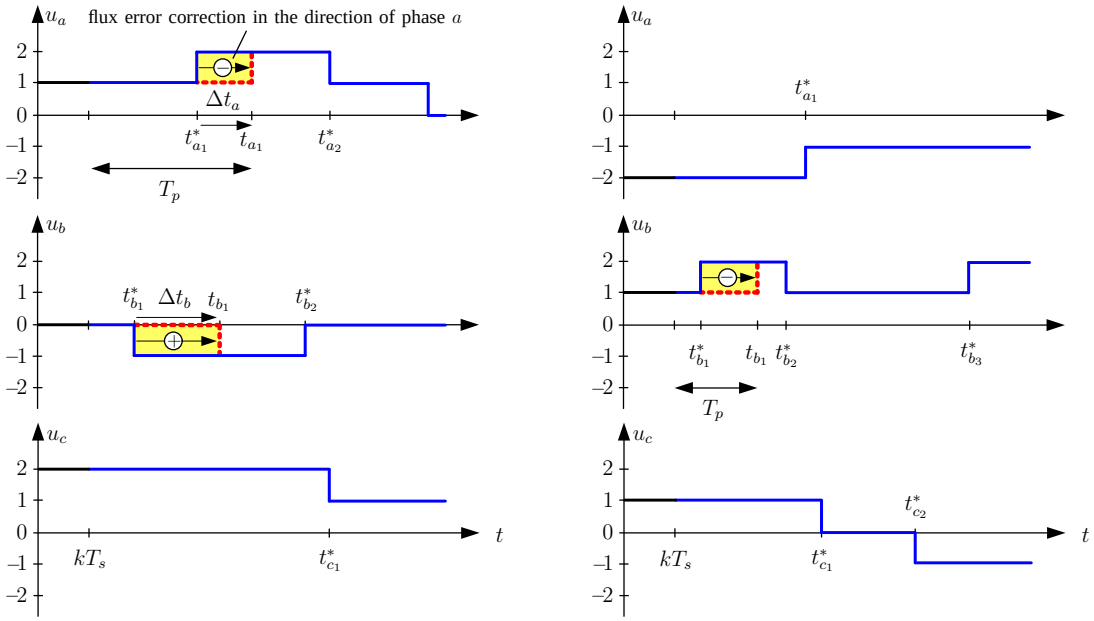
## IV. EVENT-BASED HORIZON

Recently, in [17] a new MPC algorithm was introduced for controlling MV ac drive systems exhibiting very promising performance. The proposed control scheme is a combination of MPC and optimal pulse patterns (OPPs) [31]. OPPs are calculated offline by solving an optimization problem, the objective of which is to minimize the total harmonic distortion (THD) of the machine currents in the linear and nonlinear range of the modulation index. Based on these patterns an optimal stator flux trajectory is derived [32]. The controller aims to track this trajectory so as to compensate as quickly as possible the flux error. To do so, the offline-calculated switching instants of the OPPs are read from a lookup table and modified in real-time. These modifications are the result of an optimization problem formulated in the MPC framework.

Although the formulated optimization problem proposed in [17] can be effectively solved, regardless of the length of the prediction horizon, a deadbeat (computationally and conceptually simpler) version of the strategy has been proposed [17], as well. However, the deadbeat implementation must be further refined in order to facilitate the implementation of the algorithm in a real-time system without deteriorating the closed-loop dynamic performance. To achieve this, the calculation of the *event-based* prediction horizon, initially proposed in [17], is reformulated [33]; in this way the calculations required are limited, while the superior dynamic behavior of the controller is guaranteed.

To explain how the event-based horizon is implemented, let us consider again the ANCP-5L inverter driving an IM, as shown in Fig. 8. The goal is to correct the flux error $\boldsymbol{\psi}_{s,\text{err}} = \boldsymbol{\psi}_{s,\text{ref}} - \boldsymbol{\psi}_s$, where $\boldsymbol{\psi}_{s,\text{ref}}$ is the reference stator flux vector, and $\boldsymbol{\psi}_s$ the estimated stator flux vector. To do so, the pre-calculated (nominal) switching instants $t_x^*$, $x \in \{a, b, c\}$, of the OPPs, are modified in real-time by a time interval $\Delta t_x$. Thus, a modified switching instant results: $t_x = t_x^* + \Delta t_x$. By advancing or delaying the nominal switching instants the volt-second area that the pulse sequence of each phase contributes to the flux is changed. Hence, it can be either increased or decreased depending on the direction of the modification and the switching transition [17] (Fig. 13).

The event-based horizon depends on the nominal and modified switching instants. First, the two future nominal switching instants, $t_{act_1}$ and $t_{act_2}$, which are closest to $t_0 = kT_s$ are identified. Then the phases that are involved in the flux error correction procedure are found. If the two switching instants that follow $t_0$ occur in different phases, the flux error vector is projected onto these two phases. In this case, these phases are considered to be the active ones in the flux error




(a) The switching instants are the nominal values $t_{act_1} = t^*_{b_1}$ and $t_{act_2} = t^*_{a_1}$; they are shifted to $t_{b_1}$ and $t_{a_1}$, respectively within the horizon $T_p$, shown by the black arrow. At $t = t_{a_1}$, the horizon is reevaluated.

(b) The switching instants are the nominal values $t_{act_1} = t^*_{b_1}$ and $t_{act_2} = t^*_{b_2}$; instant $t^*_{b_1}$ is shifted to $t_{b_1}$ within the horizon $T_p$, shown by the black arrow. At $t = t_{b_1}$, the horizon is reevaluated.

Fig. 13: The MPC controller is activated at time instant $kT_s$ and it modifies pre-calculated switching instants of a three-phase, five-level pulse pattern.

correction procedure and come in pairs, i.e. $\{a, b\}$, $\{b, c\}$ or $\{c, a\}$. On the other hand, if both switching instants $t_{act_1}$ and $t_{act_2}$ occur in the same phase, then switching instants in a single active phase, $a$, $b$ or $c$, are available to reduce the flux error. For example, in Fig. 13(a), the two active switching instants $t_{act_1} = t^*_{b_1}$ and $t_{act_2} = t^*_{a_1}$ are in different phases (in phases $b$ and $a$, respectively). In Fig. 13(b), on the other hand, only one phase is involved in the flux error correction procedure, since both active switching instants $t_{act_1} = t^*_{b_1}$ and $t_{act_2} = t^*_{b_2}$ are in phase $b$.

The length of the horizon is equal to the maximum difference between the initial sampling instant $t_0$ and the nominal or modified switching instants (Fig. 13), i.e.

$$T_p = \max\{t^*_x - t_0, t_x - t_0\}. \tag{3}$$

The selection of the event-based prediction horizon is key to the application-oriented implementation of the algorithm initially introduced in [17]. By restricting the length of the horizon to include at most two switching events a twofold task is achieved: the computational burden is significantly reduced relative to an unconstrained prediction horizon and a sufficiently long

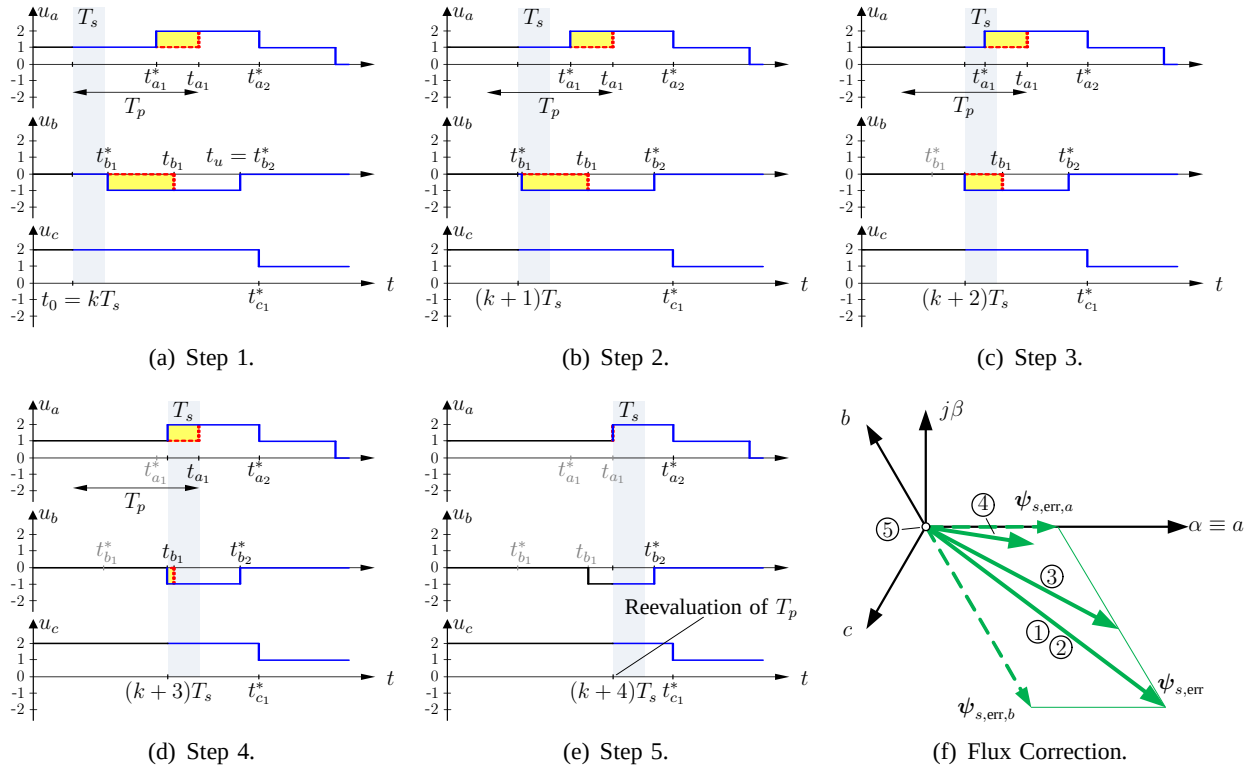
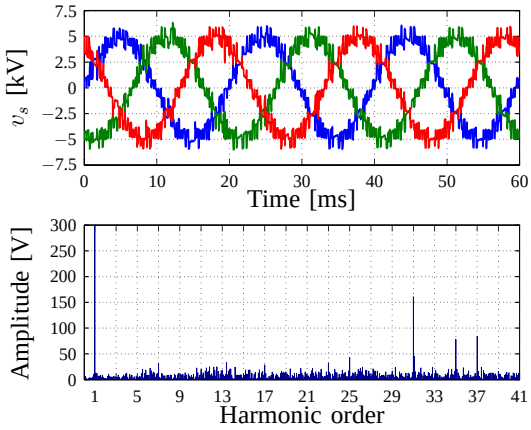



Fig. 14: Example of flux error $\boldsymbol{\psi}_{s,\text{err}}$ correction within four sampling intervals $4T_s$. The circled numbers in (f) correspond to the flux error compensation steps shown in (a)–(e).

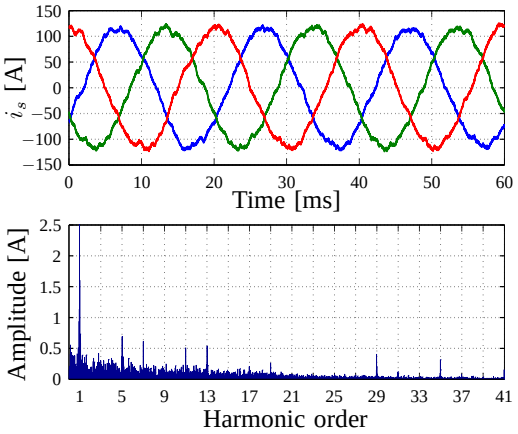
prediction horizon is achieved to effectively correct the flux error. Therefore, the implementation of the algorithm [33] on a standard microprocessor or field-programmable gate array (FPGA) device is possible. Note that in practice, the horizon is divided into equal sampling intervals of fixed length $T_s$, based on the execution sampling interval of the algorithm. When the entire horizon has been scanned, a new prediction horizon $T_p$ is evaluated according to (3) and the flux error correction procedure is repeated.

To demonstrate how the flux error correction procedure with an event-based horizon works, an illustrative example is shown in Fig. 14. The goal is to compensate the flux error $\boldsymbol{\psi}_{s,\text{err}}$ shown as bold solid lines in Fig. 14(f). In Fig. 14(a), the length of the prediction horizon $T_p$ is determined. Next, the required time modifications $\Delta t_a = -(t^*_{a_1} - t_{a_1})$ and $\Delta t_b = -(t^*_{b_1} - t_{b_1})$ are calculated within the first sampling interval $T_s$. The flux correction starts taking effect after the sampling instant $(k+1)T_s$ in Fig. 14(b) and the error is fully compensated between $(k+3)T_s$ and $(k+4)T_s$. As can be seen, in this example the flux error is eliminated in four sampling intervals (Fig. 14(a) to Fig. 14(d)). The new prediction horizon $T_p$ is determined in Fig. 14(e).

The performance of the algorithm presented in [33] was tested in the laboratory with a

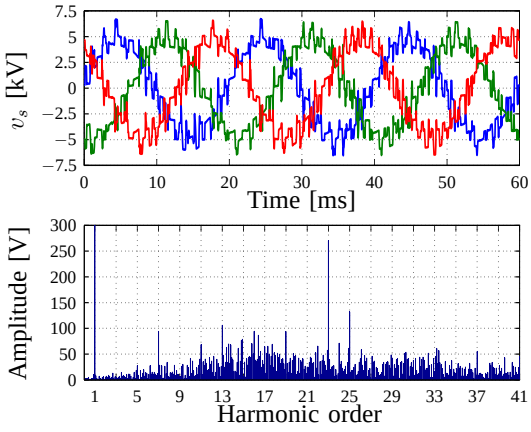



(a) Three-phase stator voltage waveforms and harmonic spectrum; the rms value of the phase voltage is 3.49 kV.

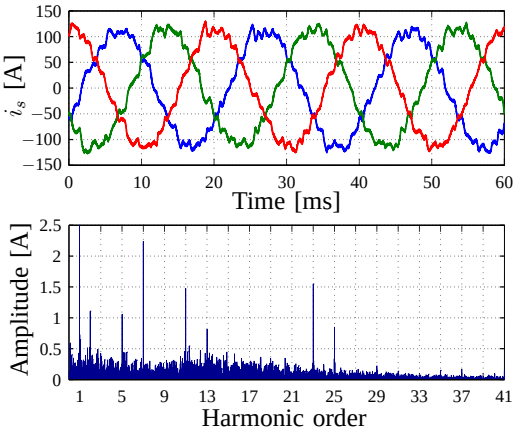



(b) Three-phase stator current waveforms and harmonic spectrum; the rms value of the phase current is 85 A. The total demand distortion (TDD) is 3.77% referred to the rated current of the controlled machine (137 A).

Fig. 15: Experimental results of the MPC strategy [33], with an event-based horizon. The drive operates in steady-state ($f_1 = 50$ Hz and 62% machine current). Optimal pulse patterns of $d = 10$ switching instants per quarter-wave are employed.




(a) Three-phase stator voltage waveforms and harmonic spectrum; the rms value of the phase voltage is 3.49 kV.




(b) Three-phase stator current waveforms and harmonic spectrum; the rms value of the phase current is 85 A. DTC produces 6.37% total demand distortion (TDD) referred to the rated current of the controlled machine (137 A).

Fig. 16: Experimental results produced by DTC in steady-state operation ($f_1 = 50$ Hz and 62% machine current). The recorded waveforms are at the same operating point and switching frequency as in Fig. 15.

sampling interval $T_s = 25\,\mu$s. The test setup consisted of a 1 MVA ACS 2000 MV drive from ABB coupled to a 6-kV, 137-A induction machine with a constant mechanical load. Stator voltage and current waveforms recorded in the experimental setup are shown in Fig. 15 while the machine was operated at 50 Hz frequency and partial load. The mean fundamental component of the stator voltage is 3.49 kV rms (Fig. 15(a)). It is shown only partially in the spectrum to emphasize the harmonic content. In Fig. 15(b), the three-phase stator current is shown. The

fundamental component of the current ($85\,\mathrm{A}$ rms) is also shown only partially and the spectrum is zoomed in to focus on the very low amplitudes of the current harmonics. The total demand distortion (TDD) of the stator currents is just $3.77\%$ referred to the rated current of the controlled machine ($137\,\mathrm{A}$).

In order to highlight the favorable performance of the proposed strategy a comparison with direct torque control (DTC) [34] was made. The drive was operated at the same operating point and switching frequency as before. The waveforms of the voltages and currents that DTC produced, as well as the respective harmonic spectra, are shown in Fig. 16. As can be seen in Fig. 16(b) DTC produces stator currents with $6.37\%$ TDD.

## V. Conclusions

Direct model predictive control (MPC) algorithms implemented for power electronics applications are computationally demanding, because a long prediction horizon is often required for guaranteed stability and improved performance of the plant. In this article, strategies that achieve long prediction horizons well within achievable levels of computational effort have been outlined and reviewed. Three techniques have been considered, namely the move blocking strategy, the extrapolation strategy, and the notion of the event-based horizon. By employing these strategies, the computational complexity is kept at a relatively low level, facilitating the execution of MPC algorithms in real-time. Moreover, by emulating long prediction horizons, the closed-loop stability margin and the performance is enhanced.

Three different case studies have been considered, which highlight the variety of problems that can be addressed with the aforementioned strategies. These case studies include a dc-dc boost converter using a move blocking strategy, an ANPC-5L ac medium voltage (MV) drive first using an extrapolation strategy, and second using an event-based horizon. These examples clearly demonstrate the effectiveness and the advantages of the aforementioned established strategies.

These techniques show just a few possibilities for making MPC feasible in power electronics, i.e. by extending the horizon with low computational penalty. Other techniques exist and will surely be developed in the future, see e.g. [35]–[37]. Actually, one of the most challenging tasks a designer faces is to develop and implement such schemes that will further improve the performance of power electronics systems, while keeping the computational burden low.

## References

[1] K. J. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning*, 2nd ed. Research Triangle Park, NC: Instrum. Soc. of Amer., 1995.

[2] M. A. Johnson and M. H. Moradi, *PID Control: New Identification and Design Methods*. London, UK: Springer, 2005.
[3] J. M. Maciejowski, *Predictive Control with Constraints*. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[4] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI: Nob Hill, 2009.
[5] M. Morari and J. H. Lee, "Model predictive control: Past, present and future," *Comput. and Chemical Eng.*, vol. 23, no. 4, pp. 667–682, May 1999.
[6] D. Q. Mayne, "Control of constrained dynamic systems," *Eur. J. of Control*, vol. 7, no. 2–3, pp. 87–99, 2001.
[7] T. Geyer, "Low complexity model predictive control in power electronics and power systems," Ph.D. dissertation, Autom. Control Lab. ETH Zurich, Zurich, Switzerland, 2005.
[8] P. Cortés, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodríguez, "Predictive control in power electronics and drives," *IEEE Trans. Ind. Electron.*, vol. 55, no. 12, pp. 4312–4324, Dec. 2008.
[9] S. Kouro, P. Cortés, R. Vargas, U. Ammann, and J. Rodríguez, "Model predictive control—A simple and powerful method to control power converters," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1826–1838, Jun. 2009.
[10] A. Linder, R. Kanchan, R. Kennel, and P. Stolze, *Model-based Predictive Control of Electric Drives*. Göttingen, Germany: Cuvillier Verlag, 2010.
[11] P. Karamanakos, "Model predictive control strategies for power electronics converters and ac drives," Ph.D. dissertation, Elect. Mach. and Power Electron. Lab. NTU Athens, Athens, Greece, 2013.
[12] C. E. García, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice—A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, Mar. 1989.
[13] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.
[14] F. Borrelli, "Discrete time constrained optimal control," Ph.D. dissertation, Autom. Control Lab. ETH Zurich, Zurich, Switzerland, 2002.
[15] D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control—Part I. The basic algorithm," *Automatica*, vol. 23, no. 2, pp. 137–148, Mar. 1987.
[16] ——, "Generalized predictive control—Part II. Extensions and interpretations," *Automatica*, vol. 23, no. 2, pp. 149–160, Mar. 1987.
[17] T. Geyer, N. Oikonomou, G. Papafotiou, and F. D. Kieferndorf, "Model predictive pulse pattern control," *IEEE Trans. Ind. Appl.*, vol. 48, no. 2, pp. 663–676, Mar./Apr. 2012.
[18] P. Karamanakos, T. Geyer, N. Oikonomou, F. Kieferndorf, and S. Manias, "Model predictive control in power electronics: Strategies to reduce the computational complexity," in *Proc. IEEE Ind. Electron. Conf.*, Vienna, Austria, Nov. 2013.
[19] R. Cagienard, P. Grieder, E. C. Kerrigan, and M. Morari, "Move blocking strategies in receding horizon control," *J. of Process Control*, vol. 17, no. 6, pp. 563–570, Jul. 2007.
[20] G. Papafotiou, T. Geyer, and M. Morari, "Optimal direct torque control of three-phase symmetric induction motors," in *Proc. IEEE Conf. Decis. Control*, Atlantis, Bahamas, Dec. 2004.
[21] T. Geyer, G. Papafotiou, and M. Morari, "Model predictive control in power electronics: A hybrid systems approach," in *Proc. IEEE Conf. Decis. Control*, Seville, Spain, Dec. 2005, pp. 5606–5611.
[22] Z. Chen, W. Gao, J. Hu, and X. Ye, "Closed-loop analysis and cascade control of a nonminimum phase boost converter," *IEEE Trans. Power Electron.*, vol. 26, no. 4, pp. 1237–1252, Apr. 2011.
[23] P. Karamanakos, T. Geyer, and S. Manias, "Direct voltage control of dc-dc boost converters using enumeration-based model predictive control," *IEEE Trans. Power Electron.*, vol. 29, no. 2, pp. 968–978, Feb. 2014.
[24] T. Geyer, G. Papafotiou, and M. Morari, "Model predictive direct torque control—Part I: Concept, algorithm and analysis," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1894–1905, Jun. 2009.
[25] G. Papafotiou, J. Kley, K. G. Papadopoulos, P. Bohren, and M. Morari, "Model predictive direct torque control—Part II: Implementation and experimental evaluation," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1906–1915, Jun. 2009.
[26] T. Geyer, "Generalized model predictive direct torque control: Long prediction horizons and minimization of switching losses," in *Proc. IEEE Conf. Decis. Control*, Shanghai, China, Dec. 2009, pp. 6799–6804.
[27] ——, "Model predictive direct current control: Formulation of the stator current bounds and the concept of the switching horizon," *IEEE Ind. Appl. Mag.*, vol. 18, no. 2, pp. 47–59, Mar./Apr. 2012.
[28] T. Geyer and S. Mastellone, "Model predictive direct torque control of a five-level ANPC converter drive system," *IEEE Trans. Ind. Appl.*, vol. 48, no. 5, pp. 1565–1575, Sep./Oct. 2012.
[29] F. Kieferndorf, P. Karamanakos, P. Bader, N. Oikonomou, and T. Geyer, "Model predictive control of the internal voltages of a five-level active neutral point clamped converter," in *Proc. IEEE Energy Convers. Congr. Expo.*, Raleigh, NC, Sep. 2012, pp. 1676–1683.
[30] F. Kieferndorf, M. Basler, L. A. Serpa, J.-H. Fabian, A. Coccia, and G. A. Scheuer, "A new medium voltage drive system based on ANPC-5L technology," in *Proc. IEEE Int. Conf. Ind. Technol.*, Viña del Mar, Chile, Mar. 2010, pp. 643–649.
[31] H. S. Patel and R. G. Hoft, "Generalized techniques of harmonic elimination and voltage control in thyristor inverters: Part I—Harmonic elimination," *IEEE Trans. Ind. Appl.*, vol. IA-9, no. 3, pp. 310–317, May 1973.
[32] J. Holtz and N. Oikonomou, "Synchronous optimal pulsewidth modulation and stator flux trajectory control for medium-voltage drives," *IEEE Trans. Ind. Appl.*, vol. 43, no. 2, pp. 600–608, Mar./Apr. 2007.
[33] N. Oikonomou, C. Gutscher, P. Karamanakos, F. Kieferndorf, and T. Geyer, "Model predictive pulse pattern control for the five-level active neutral point clamped inverter," in *Proc. IEEE Energy Convers. Congr. Expo.*, Raleigh, NC, Sep. 2012, pp. 129–136.

[34] I. Takahashi and T. Noguchi, "A new quick-response and high-efficiency control strategy of an induction motor," *IEEE Trans. Ind. Appl.*, vol. IA-22, no. 5, pp. 820–827, Sep. 1986.
[35] T. Geyer, "Computationally efficient model predictive direct torque control," *IEEE Trans. Power Electron.*, vol. 26, no. 10, pp. 2804–2816, Oct. 2011.
[36] T. Geyer and D. E. Quevedo, "Multistep direct model predictive control for power electronics—Part 1: Algorithm," in *Proc. IEEE Energy Convers. Congr. Expo.*, Denver, CO, Sep. 2013, pp. 1154–1161.
[37] ——, "Multistep direct model predictive control for power electronics—Part 2: Analysis," in *Proc. IEEE Energy Convers. Congr. Expo.*, Denver, CO, Sep. 2013, pp. 1162–1169.