Computational Analysis of the Long Horizon FCS-MPC Problem for Power Converters

Eduardo Zafra, Sergio Vazquez, *Fellow, IEEE,* Tobias Geyer, *Fellow, IEEE,* Ricardo P. Aguilera, *Member, IEEE,* Emilio Freire, and Leopoldo G. Franquelo, *Life Fellow, IEEE*

***Abstract*—Long prediction horizon finite control set model predictive control (LPH-FCS-MPC) for power converters can be reformulated as a box-constrained integer-least squares (ILS) problem to find the optimal control action without requiring an exhaustive search. Instead, the solution can be found by means of a sphere decoding method that still presents several intricacies regarding its complexity and its variable computational cost. This paper provides a study of the computational behavior of this approach. Special emphasis is placed on how the generator matrix is calculated, either as a lower or an upper triangular structure. This choice decides whether the switching sequences are explored forward- or backward-in-time during the optimization process. In this work, it is explained how this selection holds a great impact on the computational burden of the algorithm. Similarly, it is also analyzed how the tuning of the FCS-MPC and system parameters also drastically impacts the computational cost.**

***Index Terms*—Predictive control, digital control, three-phase DC-AC inverters.**

List of Acronyms

AC	Alternating Current
BTE	Backward-time Exploration
CN	Condition Number
DF	Depth-First
FCS-MPC	Finite Control Set - MPC
FTE	Forward-time Exploration
ILS	Integer-Least Squares
LPH-FCS-MPC	Long Prediction Horizon - FCS-MPC
LT	Lower Triangular
MPC	Model Predictive Control
SDA	Sphere Decoding Algorithm
TAO	Time-Ascending Order
TDO	Time-Descending Order
THD	Total Harmonic Distortion
UT	Upper Triangular

The authors gratefully acknowledge the financial support provided by the project PID2020-115561RB-C31 funded by MCIN/AEI/10.13039/501100011033 and the project TED2021-130613B-I00 funded by MCIN/AEI/10.13039/501100011033 and by the "European Union NextGenerationEU/PRTR". Ricardo P. Aguilera's work is supported by the Australian Government through the Australian Research Council (Discovery Project No. DP240102646).

Eduardo Zafra is with EKS Energy, a Hitachi Group company, Av. Camas, 28, 41110 Bollullos de la Mitacion, Seville, Spain (e-mail: ezafra1@us.es).

Sergio Vazquez and Leopoldo G. Franquelo are with the Laboratory of Engineering for Energy and Environmental Sustainability, Universidad de Sevilla, Sevilla, C.P. 41092 Spain (e-mail: sergi@us.es, lgfranquelo@ieee.org).

Tobias Geyer is with ABB System Drives, 5300 Turgi, Switzerland (e-mail: t.geyer@ieee.org).

Ricardo P. Aguilera is with the School of Electrical and Data Engineering, University of Technology Sydney, Broadway, NSW 2007, Australia (e-mail: raguilera@ieee.org).

Emilio Freire is with the Department of Applied Mathematics II, Universidad de Sevilla, Sevilla, C.P. 41092 Spain (e-mail: efrem@us.es).

I. Introduction

MODEL predictive control (MPC) is becoming a popular strategy in power electronics [1]. Although the application to the control of power converters has been delayed compared with other industrial processes where MPC has enjoyed a key role for decades, the development of increasingly powerful digital control platforms has allowed the successful development of MPC strategies, such as finite control set - MPC (FCS-MPC) [2]. FCS-MPC is a control method in which an integer optimization problem is solved every control interval to find the optimal control input. This is done according to a quadratic cost function that quantifies the suitability of each control outcome to accomplish the control objectives. One key aspect of FCS-MPC is that the control inputs are restricted to the discrete set that corresponds to the switch positions or voltage levels of the power converter [3]. In this sense, FCS-MPC is a direct control method as it prescinds from a modulation stage [4]. In this way, sampling and switching frequencies are decoupled, allowing one to increase the control bandwidth to obtain a fast dynamic response compared to indirect control methods. Also, it causes the switching frequency to be variable. For this reason, the average switching frequency $\bar{F}_{\mathrm{sw}}$ is typically calculated as a measurement of the switching effort in FCS-MPC [1].

Essential to any kind of MPC concept is the notion of prediction. To evaluate possible control outcomes of the system, one needs a discrete prediction model to compute the evolution of the system state variables at future sampling instants. The number of time steps considered in the predictions is the prediction horizon length (N_{p}). In general, selecting a long N_{p} allows the controller to make well-informed decisions and, thus, improve performance [5]. For instance, it is known that longer N_{p} helps the controller to avoid switching sequences that will excite resonances in the system, or improve the harmonic quality per switching effort [6]. In practice, N_{p}, together with the sampling and switching frequencies define the prediction window and the granularity of switching, which are crucial magnitudes as they represent the actual amount of time considered in the predictions and the accuracy in time of the problem discretization, respectively. As such, the tuning of these parameters is highly coupled and is an intricate challenge where performance, efficiency, and computational complexity are involved. This problem is thoroughly studied in [5].

Since the number of possible combinations of members in the FCS increases exponentially with N_{p}, solving the LPH-FCS-MPC through exhaustive enumeration is highly inefficient

if not completely intractable. The main existing solutions rely on the reformulation of the optimization problem as an equivalent box-constrained integer least-squares (ILS) problem [7], [8]. This is achieved through algebraic manipulation of the cost function, where the Hessian matrix of the quadratic form is decomposed by Cholesky's factorization. The obtained factor is a generator matrix $\boldsymbol{H}$ which defines a lattice where the different candidates lie. In this form, the optimal candidate can be found by branch-and-bound methods carried out through a connected graph in tree form. For example, with a sphere decoding algorithm (SDA), it is possible to find the optimal solution without exploring the entire set of candidates. Thus, the computational burden is significantly reduced [9]. A survey of the main techniques, applications, and future works of long prediction horizon FCS-MPC (LPH-FCS-MPC) can be found in [10]. Despite the advantages of the ILS approach, the computational complexity is still considerable and represents an important challenge.

This work will offer an in-depth computational analysis of the LPH-FCS-MPC problem. The goal of this study is twofold: To provide a better characterization of these techniques in regards to its computational behavior and to facilitate a more widespread adoption by helping reduce the computational cost. Two topics will be specially addressed. First, the influence of different control and system parameters on the computational cost will be studied. In particular, it will be shown how the conditioning metrics of the lattice generator matrix are essential to explain this influence. Second, it will be studied how the calculation of the lattice generator matrix as an upper- or lower-triangular structure is also crucial, as it defines the exploration mode of the search tree. Forward-in-time exploration (FTE) and backward-in-time exploration (BTE) are both possible and both provide the same optimal solution. However, the impact in terms of computational performance is large. Most works in the literature use either an FTE or a BTE mode without a proper analysis to support this decision. The main differences in formulation will be highlighted. The possibility to define switching sequences in time-ascending order (TAO) or in a time-descending order (TDO) will also be addressed in the presented study. Then, different advanced SDA techniques proposed in the literature will be considered in the comparison to determine which formulation is more adequate. This paper is structured as follows. In Section II, the control formulation is presented. Typical algorithms to solve the optimization problem are presented in Section III. Theoretical concepts regarding the complexity of lattice search problems are presented in Section IV. Simulation and experimental results are carried out in Section V. Section VI offers the conclusions to this article.

II. FCS-MPC FORMULATION

Consider a system described by a a discrete-time state-space representation with sampling frequency f_s such as:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k; \quad \boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k, \tag{1}$$

where k is the discrete time step. Vector $\boldsymbol{x} \in \mathbb{R}^n$ is the system state variable, $\boldsymbol{y} \in \mathbb{R}^{n_y}$ is the system output, and $\boldsymbol{u} \in \mathbb{V}^{n_u}$ is the system input, where $\mathbb{V} \subset \mathbb{Z}$ restricts the input to a set of integer switch positions. For the experimental tests carried out in this paper, a system plant consisting of a two-level three-phase voltage source inverter connected to an LC filter of values L_f and C_f will be considered. In this plant, the FCS is defined by $\mathbb{V} = \{0, 1\}$ and $n_u = 3$. A schematic of the system is shown in Fig. 1.

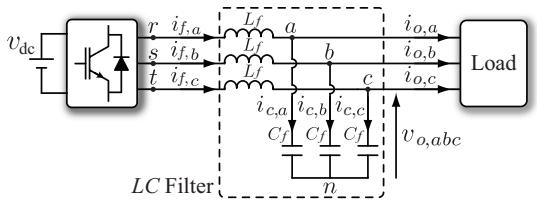


Fig. 1. Two-level three-phase voltage source inverter with output LC filter.

The setup is operated as an uninterruptible power supply system through a voltage controller defined in the stationary $\alpha\beta$-frame [11], [12]. The controller seeks to impose sinusoidal AC three-phase voltages at 50 Hz at the LC filter output terminals where a resistive inductive load is connected[1]. Further information on how to extract the prediction model from such system can be found in [13], [14].

By successively computing (1), a prediction of the system state at the next sampling instant, for a given $\boldsymbol{u}_k$, can be obtained. It is considered that the control outcome is represented by $\boldsymbol{y}$, thus, the output references are defined by $\boldsymbol{y}^\star$. For the long horizon problem, it is convenient to define sequence variables. In this way, $\boldsymbol{u}_k$ inputs within N_p are appended in a stacked vector in a time-ascending order (TAO), defining the switching sequence $\boldsymbol{U}_k \in \mathbb{V}^{n_u N_\text{p}}$ as follows:

$$\boldsymbol{U}_k \triangleq \left[\ (\boldsymbol{u}_k)^\mathsf{T} \ (\boldsymbol{u}_{k+1})^\mathsf{T} ... \ (\boldsymbol{u}_{k+N_\text{p}-1})^\mathsf{T} \ \right]^\mathsf{T}. \tag{2}$$

Similarly, the output sequence $\boldsymbol{Y}_k \in \mathbb{R}^{n_y N_\text{p}}$ and output reference sequence $\boldsymbol{Y}_k^* \in \mathbb{R}^{n_y N_\text{p}}$ are also established.

Regarding the control objectives, the FCS-MPC regulates outputs along their references while also considering the control effort. Thus, a standard FCS-MPC cost function is:

$$g_k = \sum_{\ell=k}^{k+N_\text{p}-1} \|\boldsymbol{y}_{\ell+1} - \boldsymbol{y}^\star_{\ell+1}\|_2^2 + \lambda \|\boldsymbol{u}_\ell - \boldsymbol{u}_{\ell-1}\|_2^2, \tag{3}$$

where $\lambda > 0$ is a weighting factor that adjusts the trade-off between output regulation and switching effort. Following this definition, the FCS-MPC can be described as an optimization problem that seeks the switching sequence that minimizes g:

$$\boldsymbol{U}_k^\text{opt} = \arg\min_{\boldsymbol{U}_k} g_k \tag{4a}$$

$$\text{s. t. } \boldsymbol{U}_k \in \mathbb{V}^{n_u N_\text{p}} \tag{4b}$$

$$f(\boldsymbol{x}_\ell, \boldsymbol{u}_\ell) \leq 0, \forall \ell = k, ..., k + N_\text{p} - 1 \tag{4c}$$

where (4b) are the power converter topology integer input constraints. The function f in (4c) defines other constraints that must be met during the power converter operation.

[1]Unless otherwise specified, the following parameters are selected in the tests: $L_f = 2$ mH, $C_f = 50$ μF. Load: $R = 30$ Ω and $L = 20$ mH. Dc-link: 700 V. Output reference voltage: 50 Hz, 230 Vrms (phase-to-ground).

A. ILS transformation

The ILS method in [7] requires reformulation of the cost function in matrix form. Through repeated use of (1), the expression of the system state at time step $\ell+1$ for a given switching sequence can be found:

$$\boldsymbol{x}_{\ell+1} = \boldsymbol{A}^{\ell-k+1}\boldsymbol{x}_k + [\boldsymbol{A}^{\ell-k}\boldsymbol{B} \ \ldots \ \boldsymbol{A}^0\boldsymbol{B}]\boldsymbol{U}_k. \tag{5}$$

Combining (1) and (5), it is possible to obtain the following expression for the output sequence $\boldsymbol{Y}_k$:

$$\boldsymbol{Y}_k = \boldsymbol{\Gamma}\boldsymbol{x}_k + \boldsymbol{\Upsilon}\boldsymbol{U}_k, \tag{6}$$

where $\boldsymbol{\Upsilon}$ and $\boldsymbol{\Gamma}$ are block matrices that depend on the system model and whose definition can be found in the appendix. Following (6), the cost function can be written in matrix form:

$$g_k = \|\boldsymbol{\Gamma}\boldsymbol{x}_k + \boldsymbol{\Upsilon}\boldsymbol{U}_k - \boldsymbol{Y}_k^*\|_2^2 + \lambda\|\boldsymbol{S}\boldsymbol{U}_k - \boldsymbol{E}\boldsymbol{u}_{k-1}\|_2^2, \tag{7}$$

where $\boldsymbol{S}$ and $\boldsymbol{E}$ are also found in the appendix. As can be seen, the first term maintains the information about the output regulation and the second term describes the control effort through the prediction horizon. Following the procedure described in [7], this expression can be reorganized by separating terms that depend on the optimization variable $\boldsymbol{U}_k$:

$$g_k = (\boldsymbol{U}_k)^\mathsf{T}\boldsymbol{W}\boldsymbol{U}_k + 2(\boldsymbol{F}_k)^\mathsf{T}\boldsymbol{U}_k + \boldsymbol{\epsilon}_k, \tag{8}$$

where

$$\boldsymbol{W} \triangleq (\boldsymbol{\Upsilon})^\mathsf{T}\boldsymbol{\Upsilon} + \lambda\boldsymbol{S}^\mathsf{T}\boldsymbol{S} \tag{9a}$$

$$\boldsymbol{F}_k \triangleq (\boldsymbol{\Upsilon})^\mathsf{T}(\boldsymbol{\Gamma}\boldsymbol{x}_k - \boldsymbol{Y}_k^*) - \lambda\boldsymbol{S}^\mathsf{T}\boldsymbol{E}\boldsymbol{u}_{k-1} \tag{9b}$$

$$\boldsymbol{\epsilon}_k \triangleq \|\boldsymbol{\Gamma}\boldsymbol{x}_k - \boldsymbol{Y}_k^*\|_2^2 + \lambda\|\boldsymbol{E}\boldsymbol{u}_{k-1}\|_2^2. \tag{9c}$$

The term $\boldsymbol{\epsilon}_k$ is time-varying, but it does not depend on $\boldsymbol{U}_k$. Since the FCS-MPC optimization problem in (4) is defined as *argument minima*, variables that do not depend on $\boldsymbol{U}_k$ do not alter the solution of the problem, as they merely act as an offset to the cost function. Taking advantage of this property, it is also possible to add terms to the cost function that do not alter the optimization problem. Hence, the term $\pm(\boldsymbol{F}_k)^\mathsf{T}\boldsymbol{W}^{-1}\boldsymbol{F}_k$ can be added to (8). Following the symmetric property resulting from the definition of matrix $\boldsymbol{W}$, a compact expression of the cost function can be obtained:

$$g_k = (\boldsymbol{U}_k + \boldsymbol{W}^{-1}\boldsymbol{F}_k)^\mathsf{T}\boldsymbol{W}(\boldsymbol{U}_k + \boldsymbol{W}^{-1}\boldsymbol{F}_k) + \boldsymbol{\Lambda}_k, \tag{10}$$

where $\boldsymbol{\Lambda}_k \triangleq \boldsymbol{\epsilon}_k - (\boldsymbol{F}_k)^\mathsf{T}\boldsymbol{W}^{-1}\boldsymbol{F}_k$ is defined to absorb terms that do not influence the optimization problem. The expression in (10) is a quadratic form that presents its minimum at $\boldsymbol{U}_k = -\boldsymbol{W}^{-1}\boldsymbol{F}_k$. This vector is known as the unconstrained solution to the optimization problem and is denoted as $\boldsymbol{U}_k^{\text{unc}} \in \mathbb{R}^{n_u N_\text{p}}$. In general, $\boldsymbol{U}_k^{\text{unc}} \notin \mathbb{V}^{n_u N_\text{p}}$, thus it cannot be selected as the solution to the optimization problem because it does not meet the input constraints. However, its definition can be incorporated in the cost function:

$$g_k = (\boldsymbol{U}_k - \boldsymbol{U}_k^{\text{unc}})^\mathsf{T}\boldsymbol{W}(\boldsymbol{U}_k - \boldsymbol{U}_k^{\text{unc}}) + \boldsymbol{\Lambda}_k. \tag{11}$$

This expression shows that the control algorithm must find a suitable $\boldsymbol{U}_k$ that meets the constraints and is the closest to $\boldsymbol{U}_k^{\text{unc}}$. This problem is known in mathematics as the closest vector problem [15]. However, an additional operation is needed to accommodate (11) to the closest vector problem.

B. Lattice generator matrix

The cost function must be transformed into a Euclidean distance. This can be done by decomposition of matrix $\boldsymbol{W}$ through Cholesky's algorithm. Note that any other lower-upper (LU) factorization of $\boldsymbol{W}$ where the factors are not the transpose of each other, will not allow the desired simplification.

At this point, most works in the LPH-FCS-MPC literature diverge as two different methods are used to compute the factorization, with no clear insight on why one should select either method. On one hand, several works follow the intuitive consideration of applying Cholesky factorization to obtain a nonsingular, upper triangular (UT) matrix $\boldsymbol{R}$, which is the Cholesky factor of $\boldsymbol{W}$ such that: $\boldsymbol{W} = \boldsymbol{R}^\mathsf{T}\boldsymbol{R}$. This same factorization can be expressed using the lower triangular (LT) convention as: $\boldsymbol{W} = \boldsymbol{L}_\text{s}\boldsymbol{L}_\text{s}^\mathsf{T}$, where $\boldsymbol{L}_\text{s}$ is LT. However, both conventions are completely equivalent, as the Cholesky factorization is provably unique [16]. In any case, $\boldsymbol{W}$ is factorized into the multiplication of an LT matrix by a UT matrix, in that order. To simplify (11), the transpose symbol is required to the left. Thus, the UT convention is used for this factorization method.

On the other hand, other works enunciate an alternative method in which the standard Cholesky factorization is applied on $\boldsymbol{W}^{-1}$. By inverting the resulting matrix, an LT matrix factor of $\boldsymbol{W}$, denoted as $\boldsymbol{L}$, is obtained. However, this is a different matrix than $\boldsymbol{L}_\text{s}$, defined for the LT naming convention of the standard Cholesky method. Rigorously, this alternative method originates from applying a similar algorithm to Cholesky, but doing so in reverse. As is known from linear Algebra, Cholesky's standard factorization method is an iterative algorithm that sequentially computes the coefficients of the factors starting on the upper left corner of the matrix and proceeding to the next row until reaching the bottom right corner [16]. However, a different factorization algorithm can be defined by following Cholesky's algorithm in reverse to achieve a UL factorization where both matrices are also the transpose of each other. This reverse Cholesky algorithm starts in the bottom right corner and proceeds until reaching the top left position. This method factors $\boldsymbol{W}$ into a UT matrix multiplied by an LT matrix, in that order. To have the transpose symbol in the left, the product of this factorization will be expressed as: $\boldsymbol{W} = \boldsymbol{L}^\mathsf{T}\boldsymbol{L}$. It is possible to demonstrate that applying the reverse Cholesky algorithm to $\boldsymbol{W}$ must arrive at the same result as the alternative method that applies the standard Cholesky to $\boldsymbol{W}^{-1}$, i.e., $\boldsymbol{W}$ can be factorized in:

$$\boldsymbol{W} = \underbrace{\boldsymbol{R}^\mathsf{T}\boldsymbol{R}}_{\text{Standard Cholesky (LU)}} = \underbrace{\boldsymbol{L}^\mathsf{T}\boldsymbol{L}}_{\text{Reverse Cholesky (UL)}}. \tag{12}$$

Applying standard Cholesky to $\boldsymbol{W}^{-1}$:

$$\boldsymbol{W}^{-1} = \underbrace{\boldsymbol{V}^{\mathsf{T}}\boldsymbol{V}}_{\text{LU}} \rightarrow \boldsymbol{W} = \underbrace{\boldsymbol{V}^{-1}(\boldsymbol{V}^{\mathsf{T}})^{-1}}_{\text{UL}} \rightarrow \boldsymbol{L} = (\boldsymbol{V}^{\mathsf{T}})^{-1}. \tag{13}$$

Thus, standard Cholesky on $\boldsymbol{W}^{-1}$ can be used to reach the reverse Cholesky factorization form. In any case, both factorizations can be substituted in (11) to obtain an Euclidean metric that allows one to define the LPH-FCS-MPC optimization problem in ILS-form:

$$\boldsymbol{U}_k^{\text{opt}} = \arg\min_{\boldsymbol{U}_k} g_k = \arg\min_{\boldsymbol{U}_k} \|\boldsymbol{H}\boldsymbol{U}_k - \bar{\boldsymbol{U}}_k^{\text{unc}}\|_2^2 \tag{14a}$$

$$\text{s. t. } \boldsymbol{U}_k \in \mathbb{V}^{n_u N_{\text{p}}}, \tag{14b}$$

where $\boldsymbol{H}$ is selected as $\boldsymbol{R}$ or $\boldsymbol{L}$ depending on the factorization method. From a geometrical standpoint, $\boldsymbol{H}$ represents a basis of $\mathbb{R}^{n_u N_{\text{p}}}$ that defines a linear transformation of the original CF from a hyperellipsoid form into hyperspheres aligned with the axes defined by the new basis and whose squared radii are given by the CF. Note that the bar symbol is used to identify vectors transformed to the new space. As such, $\bar{\boldsymbol{U}}_k^{\text{unc}} = \boldsymbol{H}\boldsymbol{U}_k^{\text{unc}}$ is the transformed unconstrained solution.

Since the $\boldsymbol{U}_k$ variable is not continuous, but rather a discrete set of points in space, they form what is known as a lattice. However, this original lattice lacks interest as the optimization problem will be solved in the transformed space. For this, $\boldsymbol{H}$ is known as a generator for lattice $\mathcal{L}$, a discrete subgroup in the $n_u N_{\text{p}}$-dimensional space defined by the set of points $\{\bar{\boldsymbol{U}}_k = \boldsymbol{H}\boldsymbol{U}_k : \boldsymbol{U}_k \in \mathbb{V}^{n_u N_{\text{p}}}\}$. In other words, lattice $\mathcal{L}$ is a $\mathbb{V}^{n_u N_{\text{p}}}$-linear span of $n_u N_{\text{p}}$ linearly independent vectors known as the basis vectors $\boldsymbol{h}_j$, which are the columns of $\boldsymbol{H}$.

In summary, the problem in (14) seeks the optimal lattice point $\boldsymbol{U}_k^{\text{opt}}$ with the shortest Euclidean distance in the transformed space to the target $\bar{\boldsymbol{U}}_k^{\text{unc}}$.

Computationally, the new formulation of the problem is convenient due to the triangular property of $\boldsymbol{H}$. ILS problems such as (14) can be solved with sphere decoding techniques which break down the computation of the Euclidean distance into one-dimensional subproblems. As will be explained in the next section, sphere decoding techniques constantly exploit the triangular form of the lattice matrix to evaluate partial distances without knowing the entire $\boldsymbol{U}_k$ sequence, as the zeros in matrix $\boldsymbol{H}$ will cancel terms of the product regardless of the value of the corresponding $\boldsymbol{U}_k$ components.

C. Time-Descending Order Formulation

In contrast to the standard formulation, a different definition of $\boldsymbol{U}_k$ in time-descending order (TDO) is proposed in [17]. Following the TDO, the sequences through the prediction horizon are defined backward in terms of time. Thus:

$$\boldsymbol{U}_k \triangleq \left[\ (\boldsymbol{u}_{k+N_{\text{p}}-1})^{\mathsf{T}} \ldots (\boldsymbol{u}_{k+1})^{\mathsf{T}}\ (\boldsymbol{u}_k)^{\mathsf{T}}\ \right]^{\mathsf{T}}. \tag{15}$$

Despite reversing the sequences, the fundamental relationships between the different variables is maintained in the prediction model. Thus, the LPH-FCS-MPC formulation is the same, but the block matrices defined in the appendix are flipped vertically and horizontally to accommodate the TDO formulation, as seen in [17]. According to the definition of $\boldsymbol{W}$ in (9a), the resulting matrix in the TDO formulation is then the same matrix as the TAO, but the rows and columns are flipped correspondingly. Therefore, the factorization of the TDO $\boldsymbol{W}$ has the interesting property by which standard Cholesky yields an $\boldsymbol{R}$ matrix which is in UT-form, but it is equal to the $\boldsymbol{L}$ matrix obtained by reverse Cholesky, but vertically and horizontally flipped. This can be understood by the fact that the Cholesky factors are unique. Thus, when flipping $\boldsymbol{W}$, the only factors that standard Cholesky can find are the factors found by reverse Cholesky in the original TAO formulation, but flipped to a UT form. The same logic applies to reverse Cholesky performed on the TDO $\boldsymbol{W}$, where $\boldsymbol{L}$ is a flipped version of the TAO $\boldsymbol{R}$. Fig. 2 provides a visual representation of this characterisic for clearer understanding.

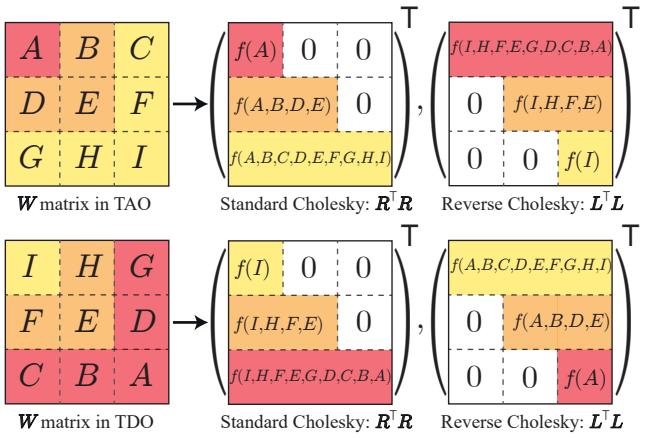


Fig. 2. TAO and TDO Cholesky factorizations. The matrices inside the parentheses represent $\boldsymbol{R}^{\mathsf{T}}$ and $\boldsymbol{L}^{\mathsf{T}}$.

III. Solution to the ILS Problem

Solving the ILS problem involves an enumeration of lattice points to find the closest one to the target $\bar{\boldsymbol{U}}_k^{\text{unc}}$. To avoid exhaustive enumeration of all candidates, branch-and-bound methods such as the SDA are typically employed to find a solution in fewer steps. Advanced computational analysis of the algorithm has derived the conclusion that, while a polynomial upper bound for the algorithm's complexity can be found for problems of relatively small dimension, the worst-case complexity is, in general, exponential. Therefore, a significant amount of research has been devoted to the definition of more efficient sphere decoding techniques for different applications [10], [18], [19].

A. Basic Concepts

The SDA defines an enumeration strategy of candidates in the space bounded by a given initial hypersphere $\mathcal{S}^{\text{ini}}$ in the transformed space. This initial hypersphere (of radius ρ^{ini}) is centered at the unconstrained solution $\bar{\boldsymbol{U}}_k^{\text{unc}}$ and is characterized by containing the initial candidate $\boldsymbol{U}_k^{\text{ini}}$. The definition of the initial hypersphere is as follows:

$$\mathcal{S}^{\text{ini}} \triangleq \left\{\boldsymbol{U}_k^{\text{ini}} : (\rho_k^{\text{ini}})^2 = \|\boldsymbol{H}\boldsymbol{U}_k^{\text{ini}} - \bar{\boldsymbol{U}}_k^{\text{unc}}\|_2^2\right\}, \tag{16a}$$

$$\text{s. t. } \|\boldsymbol{H}\boldsymbol{U}_k^{\text{opt}} - \bar{\boldsymbol{U}}_k^{\text{unc}}\|_2^2 \leq \|\boldsymbol{H}\boldsymbol{U}_k^{\text{ini}} - \bar{\boldsymbol{U}}_k^{\text{unc}}\|_2^2. \tag{16b}$$

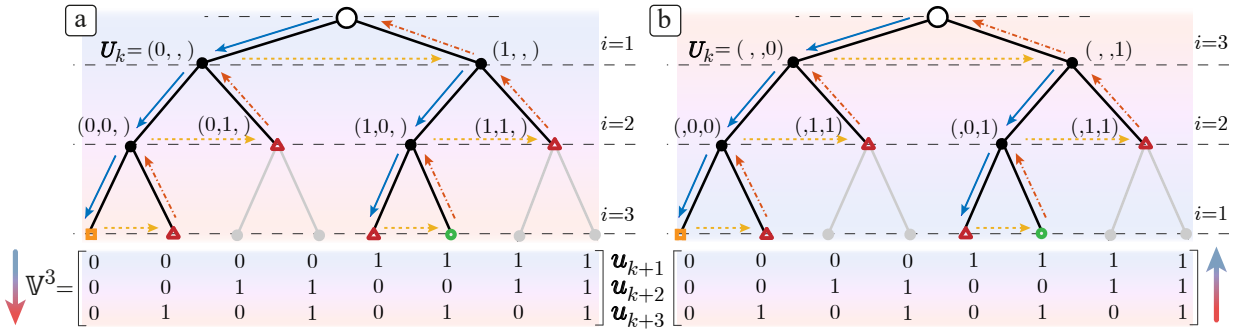


Fig. 3. Three-layer Conventional SDA search tree for $\mathbb{V} = \{0, 1\}$, $n_u = 1$ and $N_\mathrm{p} = 3$. Basic SDA operations: "forward" (solid blue arrows), "sidetracking" (dashed yellow arrows), and "backtracking" (dashed red arrows). a) Reverse Cholesky (FTE mode). b) Standard Cholesky (BTE mode).

Once an initial candidate $\boldsymbol{U}_k^{\mathrm{ini}}$ has been selected, the search stage may begin. The algorithm enumerates different candidates exploring one dimension of the input sequence in each step. For this, the integer components of each possible $\boldsymbol{U}_k$ candidate are represented as vertices of a connected graph in tree form as can be seen in Fig. 3 for a TAO formulation.

Here, the relevance of selecting either $\boldsymbol{R}$ or $\boldsymbol{L}$ as $\boldsymbol{H}$ can be observed, as it defines the exploration mode of the tree. This is because to apply the sphere decoding principle, the search must start at the row index of the generator matrix in which all the coefficients are zero except for the main diagonal. Thus, the LT form follows the FTE mode, whereas the UT form uses the BTE mode. The FTE mode explores candidates starting from the first time step, then advancing to future time steps. On the contrary, the nodes in the BTE mode are enumerated backward in terms of time. This fact does not impact the optimal solution of the optimization problem, but it can be relevant from a computational standpoint, as will be shown. Note that if a TDO is selected, the relationship is reversed, as the tree is explored in FTE mode with a UT generator matrix, and the BTE mode must be followed if the LT matrix is selected.

To search lattice points, an enumeration strategy of vertices in the connected graph, i.e. tree nodes, must be defined. Tree nodes can be identified by two integer variables, a vertical coordinate, i.e. the tree layer $i : i \in [1, n_u N_\mathrm{p}]$ and a horizontal coordinate, i.e. the switch position variable $j = \boldsymbol{U}_k(i) : j \in \mathbb{V}$. Unequivocal identification of a node is given by the partial sequence $\boldsymbol{U}_k(1 : i) = [\boldsymbol{U}_k(1) \quad \boldsymbol{U}_k(2) \quad \dots \quad \boldsymbol{U}_k(i)]^\mathsf{T}$ in a tree defined as FTE, or $\boldsymbol{U}_k(i : 3N_\mathrm{p}) = [\boldsymbol{U}_k(i) \quad \boldsymbol{U}_k(i+1) \quad \dots \quad \boldsymbol{U}_k(n_u N_\mathrm{p})]^\mathsf{T}$ in a BTE mode tree. Nodes are explored by updating the values (i, j) according to a predefined strategy.

Along with the enumeration of nodes, the Euclidean metric defined in (14) is computed. However, since nodes are generally partial candidates, this calculation is performed incrementally as each node in the tree represents a new component in the sequence for which a partial squared radius can be computed and accumulated to the total. To carry out this process, it is convenient to define an $n_u N_\mathrm{p}$-dimensional array $\boldsymbol{d}^2$ that stores the cumulative sum of the partial squared radii in each layer. In the TAO formulation, if generator matrix $\boldsymbol{H} = \boldsymbol{L}$ (FTE mode) is selected, the algorithm computes:

$$\boldsymbol{d}^2(i) = \|\boldsymbol{H}(i, 1 : i)\boldsymbol{U}_k(1 : i) - \bar{\boldsymbol{U}}_k^{\mathrm{unc}}(i)\|_2^2 + \boldsymbol{d}^2(i-1), \quad (17)$$

where $\boldsymbol{H}(i, 1 : i)$ is a vector formed by the first i elements of the ith row of matrix $\boldsymbol{H}$. Note that $\boldsymbol{d}^2(0) = 0$ and $\boldsymbol{d}^2(n_u N_\mathrm{p}) = \rho^2$ is the Euclidean distance of an entire sequence. If $\boldsymbol{H} = \boldsymbol{R}$, then the partial radius is computed as:

$$\boldsymbol{d}^2(i) = \|\boldsymbol{H}(i, i : n_u N_\mathrm{p})\boldsymbol{U}_k(i : n_u N_\mathrm{p}) - \bar{\boldsymbol{U}}_k^{\mathrm{unc}}(i)\|_2^2 + \boldsymbol{d}^2(i+1), \quad (18)$$

where $\boldsymbol{H}(i, i : n_u N_\mathrm{p})$ denotes the last i components of row i of matrix $\boldsymbol{H}$, $\boldsymbol{d}^2(n_u N_\mathrm{p} + 1) = 0$, and $\boldsymbol{d}^2(1) = \rho^2$.

B. Enumeration Strategies

Search strategies applied to tree graphs as the one presented in Fig. 3 are typically classified as depth-first or breadth-first [20]. Depth-first strategies prioritize exploring as far down a branch of the search space as possible before moving to a neighboring node in the same layer. In contrast, breadth-first strategies explore neighbor nodes in the selected layer before advancing to a different layer. The conventional SDA follows a depth-first enumeration strategy where the partial Euclidean distance of the current node is compared to that of the incumbent solution, which is initialized to $(\rho_k^{\mathrm{ini}})^2$.

The search procedure can be decomposed into three basic operations: "forward", "sidetracking", and "backtracking", appearing in order of priority. In each node, the partial Euclidean distance is compared to the cost of the incumbent solution. If it is smaller, a "forward" operation is performed to continue exploring the current branch. Otherwise, the SDA opts for "sidetracking", or "backtracking" in the case of a dead end (bottom layer or suboptimal candidate). If a partial squared radius exceeds the current best, the corresponding node is guaranteed to be suboptimal. Thus, the subsequent branch can be pruned without assessing the entire $\boldsymbol{U}_k$ sequence.

By performing early discards of candidates, the SDA can achieve superior performance to an exhaustive enumeration method. However, it is still limited in terms of achievable performance due to its high computational variability and search speed limitations. As a consequence, the SDA typically needs the definition of an early termination criterion that guarantees the bounding of the search process for a given available calculation time regardless of optimal termination. This criterion is typically defined for a specific number of explored nodes $N_{\mathrm{nod}}^{\max}$ [21]. In general, real-time control can

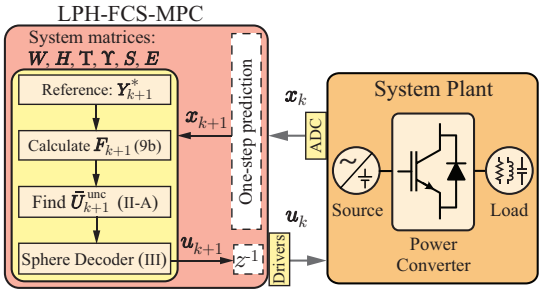


Fig. 4. Block diagram of LPH-FCS-MPC algorithm.

be ensured, but a certain degree of suboptimality is introduced in the selected solutions. Due to this, recent research seeks to define more powerful search techniques that can avoid the effects of suboptimality to a greater extent [10]. In [14], a parallel SDA was proposed by which the SDA tree is explored concurrently in depth-first fashion.

A different approach is followed in breadth-first algorithms such as the K-best SDA proposed in [22]. Breadth-first techniques prioritize horizontal enumeration of nodes, performing comparisons between neighbor nodes at the same tree level, and developing a K_b number of nodes, which are the most promising. This strategy can deliver good solutions with fixed computational costs and a low number of evaluated nodes. However, optimality is not guaranteed.

In general, different search strategies can be defined to solve the optimization stage. Regardless of their inherent characteristics, they are still affected by the complexity of the optimization problem in terms of convergence speed and suboptimality. The LPH-FCS-MPC control method based on the ILS reformulation is summarized in Fig. 4.

IV. Complexity of Lattice Search Problems

This section studies the properties of lattice generator matrix $\boldsymbol{H}$. A key concept to understand is that a single lattice $\mathcal{L}$ can be represented by different bases that generate the same lattice. However, not all bases will show identical computational behavior. In optimization problems, such as the ILS, different problem definitions are better conditioned than others. The conditioning of a problem relates to the sensitivity of the solution of said problem to changes in the input data. Typically, a well-conditioned problem is more numerically stable than an ill-conditioned problem, where the correct solution becomes harder to find.

In lattice search problems, conditioning is related to the way in which the basis vectors are disposed to facilitate the search in terms of convergence speed. For this, it is desirable that the basis vectors are pairwise as orthogonal as possible [23]. Conveniently, it is possible to encapsulate this property in different metrics rather than calculating every angle one by one. For instance, it is known that when all basis vectors meet this requirement, the determinant of the matrix $\boldsymbol{H}$ is equal to the product of the lengths of the basis vectors [23]. In any case, the determinant is less than or equal to this product. Thus, the extent to which this equality is not met can be used to measure how far from being orthogonal the basis vectors are. This property is captured by Hadamard's inequality:

$$|\det(\boldsymbol{H})| \leq \prod_{j=1}^{n} \|\boldsymbol{h}_j\|. \tag{19}$$

Another typical metric is the condition number (CN) of the basis matrix $\boldsymbol{H}$. The CN is associated with solving the linear system $\boldsymbol{H}\boldsymbol{x} = \boldsymbol{b}$ and measures the ratio of the relative error in the solution $\boldsymbol{x}$ to a relative error in $\boldsymbol{b}$ or $\boldsymbol{H}$ [24]. In particular, a perturbation of $\boldsymbol{b}$ may produce a relative change in $\boldsymbol{x}$ in the same order of magnitude as the CN. Thus, well-conditioned problems tend to exhibit lower CN values. The CN can be calculated as follows:

$$\mathrm{CN} = \|\boldsymbol{H}\|\|\boldsymbol{H}^{-1}\|. \tag{20}$$

A. Equivalence of Lattices

To study the impact on the computational behavior of the selection of either $\boldsymbol{R}$ or $\boldsymbol{L}$ as the lattice generator, it will first be demonstrated if this selection influences the conditioning of the problem. Two lattices are called equivalent if one can be obtained from the other by rotation, reflection, scaling, or a combination of these operations [25]. As can be understood, two equivalent matrices will have the same conditioning, as the basis vectors have the same angles. As is known from basic algebra, rotation and reflection are operations represented by orthogonal matrices. Thus, it must be demonstrated that $\boldsymbol{R}$ can be obtained from $\boldsymbol{L}$ or vice versa multiplying by an orthogonal matrix. Following (12), one can left-multiply the equation by $(\boldsymbol{R}^\mathsf{T})^{-1}$ and right-multiply by $\boldsymbol{L}^{-1}$ to obtain:

$$\boldsymbol{R}\boldsymbol{L}^{-1} = (\boldsymbol{R}^\mathsf{T})^{-1}\boldsymbol{L}^\mathsf{T} \rightarrow \boldsymbol{R}\boldsymbol{L}^{-1} = (\boldsymbol{L}\boldsymbol{R}^{-1})^\mathsf{T}. \tag{21}$$

Defining $\boldsymbol{P} \triangleq \boldsymbol{R}\boldsymbol{L}^{-1}$, its inverse is $\boldsymbol{P}^{-1} = \boldsymbol{L}\boldsymbol{R}^{-1}$. Substituting into (21), it is possible to write: $\boldsymbol{P} = (\boldsymbol{P}^{-1})^\mathsf{T}$. Therefore, matrix $\boldsymbol{P}$ is an orthogonal matrix. Since the definition of $\boldsymbol{P}$ allows one to express $\boldsymbol{R} = \boldsymbol{P}\boldsymbol{L}$, clearly basis $\boldsymbol{R}$ is the result of an orthogonal change from basis $\boldsymbol{L}$. Thus, the angles and the norms of the basis vectors (columns) of $\boldsymbol{R}$ and $\boldsymbol{L}$ are identical. As a consequence of this result, the conditioning metrics must be equal for both bases.

B. Conditioning evaluation

In Fig. 5, Hadamard's inequality and the CN are assessed for both Cholesky factorizations in function of the tuning parameter λ. Several conclusions can be drawn from this result. First, both selected metrics seem to describe the problem conditioning very closely. Thus, the CN will be used henceforth. Second, both factorizations produce an equally conditioned lattice generator matrix, as was expected due to their equivalence. Third, the problem conditioning is clearly affected by the selection of λ. Since this weighting factor is a penalty on the switching effort, this result implies that setting a higher switching frequency has a negative impact on the problem conditioning. Similarly, other system or control parameters can be evaluated to assess how they will influence the problem conditioning.

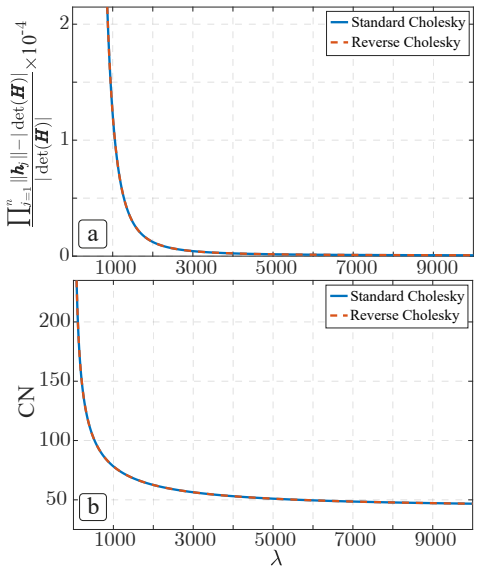


Fig. 5. Conditioning assessment of the LPH-FCS-MPC problem of size $n_u = 3$ and $N_\mathrm{p} = 7$. (a) Hadamard's inequality. (b) CN.

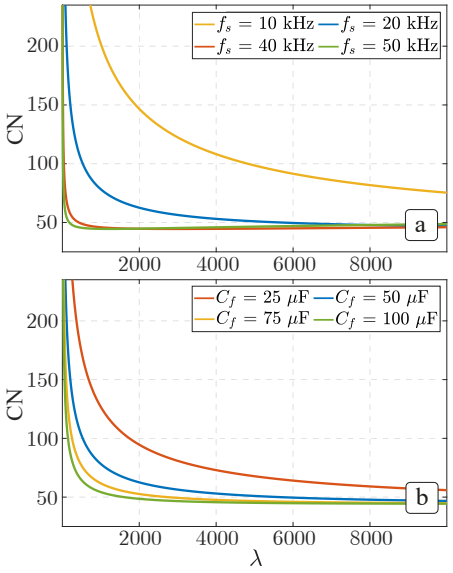


Fig. 6. Assessment of the CN for different parameters ($n_u = 3$, $N_\mathrm{p} = 7$). (a) Sampling frequency f_s. (b) Filter capacitor C_f.

In Fig. 6, conditioning metrics are presented for different sampling frequencies, showing its notable influence on the problem conditioning. In particular, a coarse discretization worsens the conditioning of the problem. The same trend is observed if the filter capacitor is smaller. This clearly highlights how physical parameters in the system and control parameters can have a large impact on the problem conditioning. Note that the results in Fig. 6, are replicated for either formulation.

C. Coefficients distribution of matrices

The equivalence demonstrated in Section IV-A implies that the norm of the columns of $\boldsymbol{R}$ and $\boldsymbol{L}$ must be identical. However, this is not the case for the rows. When evaluating the norm of the rows, the structure shown in Fig. 7 can

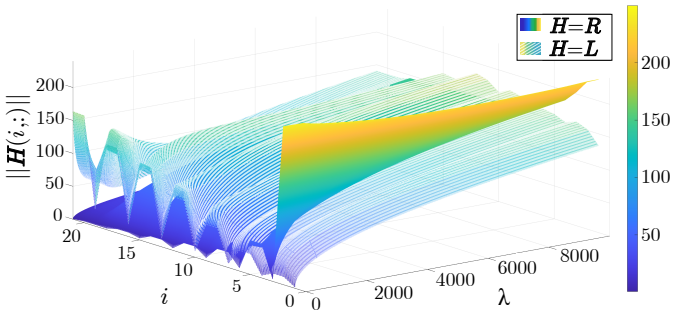


Fig. 7. Norm of the i-row vectors of matrix $\boldsymbol{H}$ for different λ and $N_\mathrm{p} = 7$.

be observed. This structure originates from the distribution of coefficients in matrix $\boldsymbol{W}$. Recall its definition in (9a). Considering the triangular nature of $\boldsymbol{\Upsilon}$, the multiplication of $\boldsymbol{\Upsilon}^\mathsf{T}\boldsymbol{\Upsilon}$ yields a matrix where the top left corner tends to have the largest numbers in absolute value, while many cancellations occur in the bottom right corner due to the zeros. Note that this structure is not much modified by the addition of the $\lambda \boldsymbol{S}^\mathsf{T}\boldsymbol{S}$ term, as this is a sparse matrix that only adds an offset to the main diagonal and to two secondary diagonals. Due to the sequential nature of the Cholesky algorithm, the factors of matrix $\boldsymbol{W}$ are computed so that any upper-left block from $\boldsymbol{R}^\mathsf{T}$ only depends on the corresponding upper-left block of $\boldsymbol{W}$, as seen in Fig. 2. Thus, $\boldsymbol{R}^\mathsf{T}$ tends to respect the coefficient distribution in terms of relative size.

This is relevant because, as seen in Section III, each step of the search procedure considers a new row of matrix $\boldsymbol{H}$. Thus, the distribution of the norms of each row defines the order in which the Euclidean distance is computed. On one hand, the FTE mode followed with $\boldsymbol{L}$ starts the search at $i = 1$. In this mode, the algorithm will find relatively large weights in the first steps. Therefore, the computed partial distances converge faster to the full Euclidean distance in the first steps. This allows the algorithm to bound the search space faster and prune more branches. On the other hand, the BTE mode followed with $\boldsymbol{R}$ starts the search at $i = n_u N_p$. However, due to the structure of $\boldsymbol{W}$, these rows hold relatively little information. Thus, the algorithm needs to explore deeper nodes to get conclusive measurements of the Euclidean distances to make good pruning decisions. In conclusion, it is hypothesized that, although the lattice conditioning is identical, the selection of the FTE mode is crucial to speed up the search stage, as will be evaluated in the next section.

V. Performance Evaluation

This section is dedicated to evaluate the concepts described in Section IV with simulation and experimental results. For this, the case study referenced in Section II is used. In general, performance will be evaluated according to: a) Optimality $Op(\%)$, defined as the percentage of time steps in a simulation at which the optimal solution has been found, b) average cost of the final solution, d_μ, which is calculated for the totality of the simulation time, and c) output voltage total harmonic distortion, THD (%), which is selected as a measure

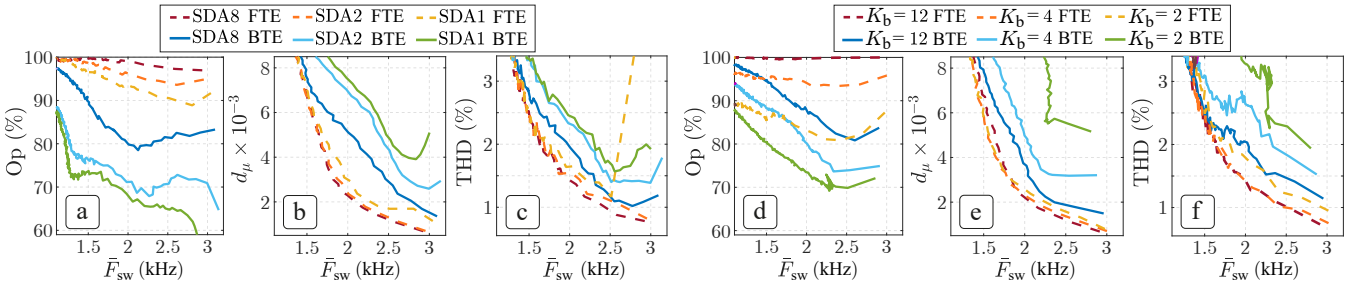


Fig. 8. Simulation results. a,d) Optimality ($Op(\%)$), b,e) average cost (d_μ) scaled dividing by 1000, and c,f) THD (%). FCS-MPC is configured with $f_s = 20$ kHz, $N_\mathrm{p} = 7$. a-c) Depth-first parallel SDA (SDAm) with m parallel blocks and $N_\mathrm{nod}^\mathrm{max}$ = 300. d-f) Breadth-first K-best SDA for different K_b values.

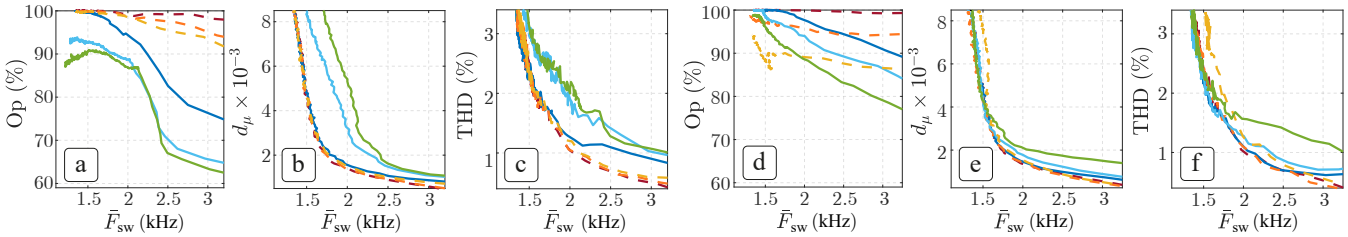


Fig. 9. Simulation results. a,d) Optimality ($Op(\%)$), b,e) average cost (d_μ) scaled dividing by 1000, and c,f) THD (%). FCS-MPC is configured with $f_s = 40$ kHz, $N_\mathrm{p} = 8$. a-c) Depth-first parallel SDA (SDAm) with m parallel blocks and $N_\mathrm{nod}^\mathrm{max}$ = 300. d-f) Breadth-first K-best SDA for different K_b values.

TABLE I
VIVADO TESTBENCH RESULTS FOR 1000 OPTIMIZATION PROBLEMS

	Total time (ms)	N^nod / blocks
SDA 1 BTE	560.21	10185438
SDA 1 FTE	86.50	1572246
SDA 4 BTE	260.81	3934207
SDA 4 FTE	43.82	534440
SDA 8 BTE	181.82	2477722
SDA 8 FTE	31.65	285566

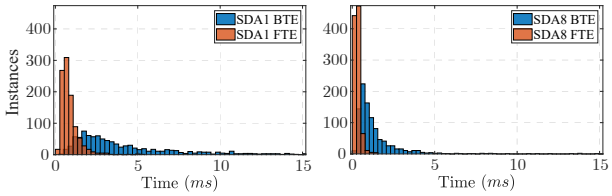


Fig. 10. Calculation time histograms for $N_\mathrm{p} = 7$ in Vivado simulator.

of controller performance to assess how search improvements translate to control outcome.

Note that by following the TDO definition in [17], a UT formulation explored in FTE is equivalent to the LT form in the original TAO definition and the results in terms of search efficiency are identical. Due to this equivalence, only results for the TAO in BTE or FTE mode will be shown. The results can be directly translated to the TDO definition by exchanging the UT and LT matrix forms.

The first test is performed in Matlab/Simulink. The plant is simulated with an LPH-FCS-MPC strategy with different search techniques and different parameters. Results are depicted in Fig. 8 and Fig. 9 in function of the resulting average switching frequency $\bar{F}_\mathrm{sw}$. As can be seen, opting for an FTE mode provides important improvements in both optimality and cost. This translates to better control outcome as the harmonic performance is also improved when using the same enumeration strategy. An exception is found in very computationally limited situations, as shown in Fig. 8-c) for the SDA1 FTE results if $\bar{F}_\mathrm{sw} > 2.5$ kHz. In this case, the FTE mode can be counterproductive because not all the switching states corresponding to the first time step may be explored, while in the BTE mode it is more likely that at least one instance of each FCS member is explored for the first time step. In any case, such a limit situation should be avoided by setting a shorter N_p or using a more powerful search strategy. Regarding control parameters, the results clearly reproduce the conditioning metrics evaluated in Section IV. For a given search strategy, a higher degree of optimality is obtained with lower $\bar{F}_\mathrm{sw}$ or higher f_s, due to the conditioning of the lattice.

A second benchmark is established in the Xilinx Vivado simulation tool, where the different algorithms are allowed to explore as many nodes as required to complete the search process. For this test, 1000 optimization problems of size $N_\mathrm{p} = 7$ are randomly generated. The initial solution is selected as a null vector in all cases to disregard the influence of the initial solution on the search stage. Through this approach, it can be measured how many nodes and how much time each algorithm will need to successfully complete all the considered problems. The results are represented in Table I. Also, calculation time histograms are presented in Fig. 10. As can be seen, FTE strategies achieve a remarkable overall reduction in calculation time.

In the third test, experiments are carried out with the laboratory prototype shown in Fig. 11 and validated in terms of the IEC 62040-3 standard for uninterruptible power supplies. In one of the experiments, a depth-first search strategy is

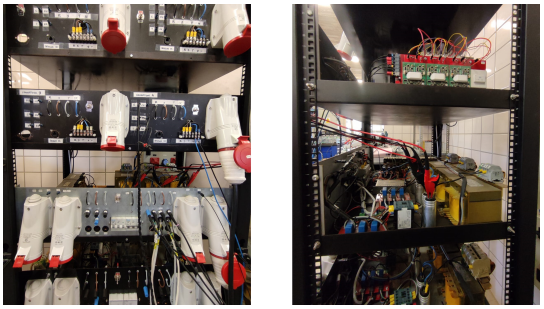


Fig. 11. Experimental test bench. Front and side view.

TABLE II
SUMMARY OF THE RESULTS

Sequence definition	TAO		TDO	
Matrix form	UT	LT	UT	LT
Exploration Mode	BTE	FTE	FTE	BTE
$\boldsymbol{H}$ calculation costs	Low	High	Low	High
Optimization stage costs	High	Low	Low	High
Application example	[14,21,22]	[3,7,9]	[17]	N/A

implemented, and it is performed for a linear load and a set of parameters. To validate a different case, a second experiment is carried out with different parameters, a breadth-first strategy, and with a nonlinear load. The resulting waveforms, along with the dynamic and harmonic levels, are illustrated in Fig. 12 and Fig. 13. As can be seen, due to the nonlinear load, the experiments in Fig. 13 show a distorted output current with non-triplen odd harmonics. However, the controller is capable of maintaining a reasonable harmonic quality in the regulated output voltage despite the harmonic-polluted current drawn by the rectifier load. Although there are important differences between both experiments, concerning the selected load, search strategy and tuning parameters, a similar conclusion can be extracted in regards to the selection of the exploration mode. Clearly, an FTE mode can help the system to achieve better THD and deviation metrics, making it easier to meet the IEC standard for a given average switching frequency.

A summary of the comparative results for the different formulations is provided in Table II.

VI. Conclusion

In this work, an analysis of the computational costs of LPH-FCS-MPC is provided. The differences between several formulations in the literature are described and analyzed. Mathematical properties of the resulting lattice generators are explored, extracting conditioning metrics in function of different system and control parameters. These properties are then assessed by simulation and experimental results. In short, switching sequences can be defined in a time-ascending order (TAO) or a time-descending order (TDO) through the prediction horizon. Depending on the selected factorization method of $\boldsymbol{W}$, the exploration mode of the tree is defined as backward-in-time exploration (BTE) or forward-in-time exploration (FTE). In any case, an upper triangular (UT) formulation avoids one matrix inversion while computing $\boldsymbol{H}$. Regarding the optimization stage, it is shown that the FTE mode is superior by almost one order of magnitude in terms of computational costs. For this reason, a TDO formulation where a UT $\boldsymbol{H}$ is obtained through standard Cholesky factorization is the best overall option in terms of calculation costs. In time-invariant systems where the matrix $\boldsymbol{H}$ can be computed offline, a TAO definition with reverse Cholesky decomposition achieves the same low computational cost. Thus, it can be favored in these situations due to its more natural formulation. In any case, it is shown that an FTE mode can remarkably help to boost the computational performance during the optimization stage regardless of the search technique. As a result, the optimization problem underlying LPH-FCS-MPC is solved more quickly, increasing the likelihood that the optimal solution is found in the available time, improving overall performance.

Appendix

The expressions for the relevant LPH-FCS-MPC block matrices in the TAO formulation are as follows:

$$\boldsymbol{\Upsilon} = \begin{bmatrix} \boldsymbol{CB} & \boldsymbol{O} & \dots & \boldsymbol{O} \\ \boldsymbol{CAB} & \boldsymbol{CB} & \dots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{CA}^{N_\mathrm{p}-1}\boldsymbol{B} & \boldsymbol{CA}^{N_\mathrm{p}-2}\boldsymbol{B} & \dots & \boldsymbol{CB} \end{bmatrix}, \boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{CA} \\ \boldsymbol{CA}^2 \\ \vdots \\ \boldsymbol{CA}^{N_\mathrm{p}} \end{bmatrix},$$

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} & \dots & \boldsymbol{O} \\ -\boldsymbol{I} & \boldsymbol{I} & \dots & \boldsymbol{O} \\ \boldsymbol{O} & -\boldsymbol{I} & \dots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{I} \end{bmatrix}, \text{ and } \boldsymbol{E} = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{O} \\ \vdots \\ \boldsymbol{O} \end{bmatrix}.$$

The expressions for the relevant LPH-FCS-MPC block matrices in the TDO formulation are as follows:

$$\boldsymbol{\Upsilon} = \begin{bmatrix} \boldsymbol{CB} & \boldsymbol{CAB} & \dots & \boldsymbol{CA}^{N_\mathrm{p}-1}\boldsymbol{B} \\ \boldsymbol{O} & \boldsymbol{CB} & \dots & \boldsymbol{CA}^{N_\mathrm{p}-2}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{CB} \end{bmatrix}, \boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{CA}^{N_\mathrm{p}} \\ \boldsymbol{CA}^{N_\mathrm{p}-1} \\ \vdots \\ \boldsymbol{CA} \end{bmatrix},$$

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{I} & \dots & \boldsymbol{O} \\ -\boldsymbol{I} & \boldsymbol{I} & \dots & \boldsymbol{O} \\ \boldsymbol{O} & -\boldsymbol{I} & \dots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \dots & \boldsymbol{I} \end{bmatrix}, \text{ and } \boldsymbol{E} = \begin{bmatrix} \boldsymbol{O} \\ \boldsymbol{O} \\ \vdots \\ \boldsymbol{I} \end{bmatrix}.$$

References

[1] P. Karamanakos and T. Geyer, "Guidelines for the design of finite control set model predictive controllers," *IEEE Trans. Power Electron.*, vol. 35, no. 7, pp. 7434–7450, 2020.

[2] S. Kouro, M. A. Perez, J. Rodriguez, A. M. Llor, and H. A. Young, "Model predictive control: Mpc's role in the evolution of power electronics," *IEEE Ind. Electronics Magazine*, vol. 9, no. 4, pp. 8–21, 2015.

[3] R. Baidya, R. P. Aguilera, P. Acuna, S. Vazquez, and H. d. T. Mouton, "Multistep model predictive control for cascaded h-bridge inverters: Formulation and analysis," *IEEE Trans. Power Electron.*, vol. 33, no. 1, pp. 876–886, 2018.

[4] J. Rodriguez, M. P. Kazmierkowski, J. R. Espinoza, P. Zanchetta, H. Abu-Rub, H. A. Young, and C. A. Rojas, "State of the art of finite control set model predictive control in power electronics," *IEEE Trans. on Industrial Informatics*, vol. 9, no. 2, pp. 1003–1016, 2013.

[5] E. Zafra, S. Vazquez, T. Dragičević, A. M. Alcaide, J. I. Leon, and L. G. Franquelo, "Prediction window selection in fcs-mpc for two-level vsi applications," *IEEE Trans. on Power Electronics*, vol. 39, no. 3, pp. 3135–3143, 2024.

[6] P. Karamanakos, E. Liegmann, T. Geyer, and R. Kennel, "Model predictive control of power electronic systems: Methods, results, and challenges," *IEEE Open Journal of Industry Applications*, vol. 1, pp. 95–114, 2020.

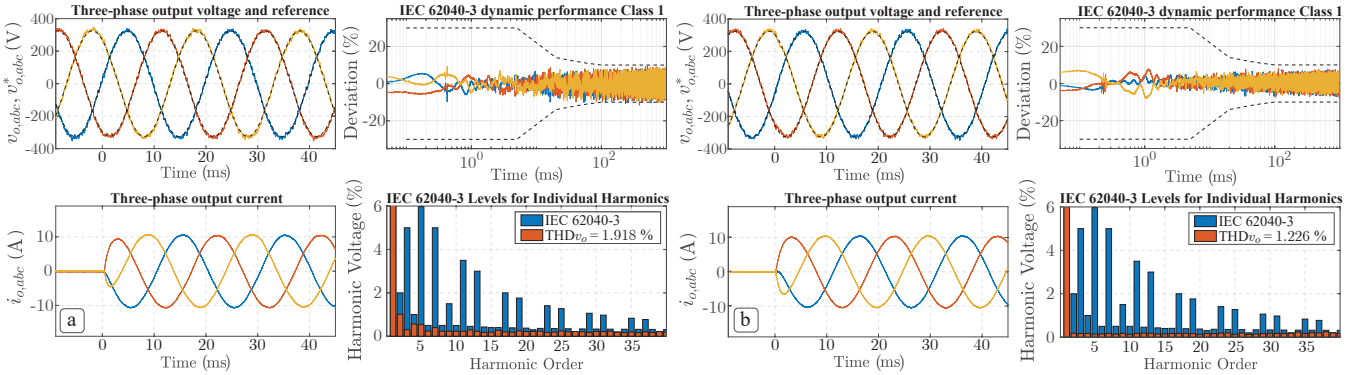


Fig. 12. Experimental results for $N_\mathrm{p} = 7$ and $f_\mathrm{s} = 20$ kHz. Parameter λ is selected to achieve $\bar{F}_\mathrm{sw} = 2$ kHz, and $N_\mathrm{nod}^\mathrm{max}$ is limited to 400. Top figures show the three-phase output voltage and reference. Bottom figures show the three-phase output current. a) SDA 2 BTE. b) SDA 2 FTE.

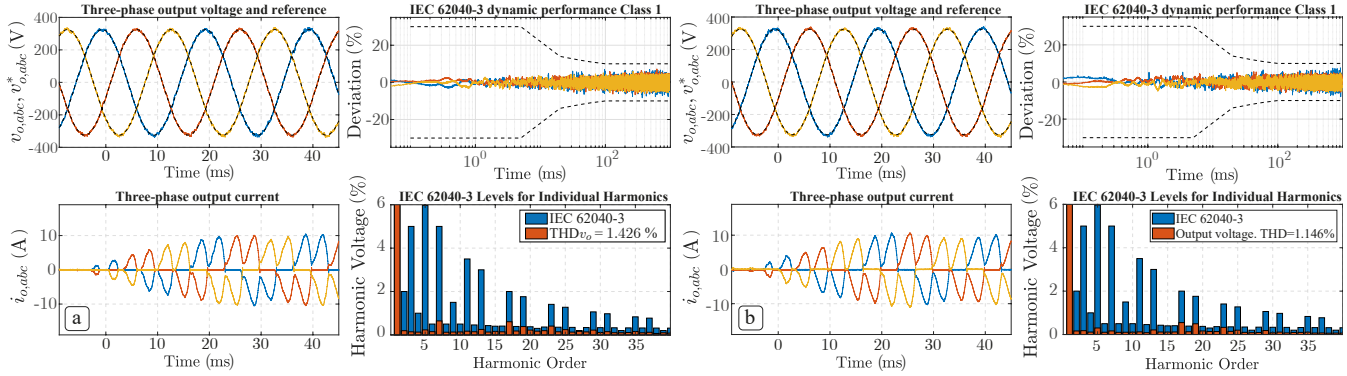


Fig. 13. Experimental results for $N_\mathrm{p} = 7$ and $f_\mathrm{s} = 25$ kHz. Parameter λ is selected to achieve $\bar{F}_\mathrm{sw} = 2.5$ kHz. The nonlinear output load is a diode rectifier with an equivalent 100 Ω resistor in the DC side. Search strategy is a K-best SDA with $K_\mathrm{b} = 8$. a) K-best BTE. b) K-best FTE.

[7] T. Geyer and D. E. Quevedo, "Multistep finite control set model predictive control for power electronics," *IEEE Trans. Power Electron.*, vol. 29, no. 12, pp. 6836–6846, 2014.

[8] R. Baidya, R. P. Aguilera, P. Acuna, T. Geyer, R. A. Delgado, D. E. Quevedo, and H. d. T. Mouton, "Enabling multistep model predictive control for transient operation of power converters," *IEEE Open Journal of the Industrial Electronics Society*, vol. 1, pp. 284–297, 2020.

[9] T. Geyer and D. E. Quevedo, "Performance of multistep finite control set model predictive control for power electronics," *IEEE Trans. Power Electron.*, vol. 30, no. 3, pp. 1633–1644, 2015.

[10] E. Zafra, S. Vazquez, T. Geyer, R. P. Aguilera, and L. G. Franquelo, "Long prediction horizon fcs-mpc for power converters and drives," *IEEE Open Journal of the Industrial Electronics Society*, vol. 4, pp. 159–175, 2023.

[11] P. Cortes, G. Ortiz, J. I. Yuz, J. Rodriguez, S. Vazquez, and L. G. Franquelo, "Model predictive control of an inverter with output *lc* filter for ups applications," *IEEE Trans. on Industrial Electronics*, vol. 56, no. 6, pp. 1875–1883, June 2009.

[12] T. Dragicevic, "Model predictive control of power converters for robust and fast operation of ac microgrids," *IEEE Trans. on Power Electron.*, vol. 33, no. 7, pp. 6304–6317, July 2018.

[13] S. Vazquez, E. Zafra, R. P. Aguilera, T. Geyer, J. I. Leon, and L. G. Franquelo, "Prediction model with harmonic load current components for fcs-mpc of an uninterruptible power supply," *IEEE Trans. Power Electron.*, vol. 37, no. 1, pp. 322–331, 2022.

[14] E. Zafra, S. Vazquez, C. Regalo Nunez, V. Baena Lecuyer, A. Marquez Alcaide, J. I. Leon, and L. Garcia Franquelo, "Parallel sphere decoding algorithm for long prediction horizon fcs-mpc," *IEEE Trans. Power Electron.*, vol. 37, no. 7, pp. 7896–7906, 2022.

[15] D. E. Knuth, *The Art of Computer Programming, Volume III: Sorting and Searching*, A. Wesley, Ed., 1973.

[16] G. Lindfield and J. Penny, *Chapter 2 - Linear Equations and Eigensystems*, fourth edition ed., G. Lindfield and J. Penny, Eds. Academic Press, 2019.

[17] A. Andersson and T. Thiringer, "Assessment of an improved finite control set model predictive current controller for automotive propulsion applications," *IEEE Trans. on Industrial Electronics*, vol. 67, no. 1, pp. 91–100, 2020.

[18] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. on Information Theory*, vol. 48, no. 8, pp. 2201–2214, 2002.

[19] E. Zafra, J. Granado, V. B. Lecuyer, S. Vazquez, A. M. Alcaide, and L. G. Leon, Jose I. Franquelo, "Computationally efficient sphere decoding algorithm based on artificial neural networks for long-horizon fcs-mpc," *IEEE Trans. on Industrial Electronics*, vol. 71, no. 1, pp. 39–48, 2024.

[20] Zhan Guo and P. Nilsson, "Algorithm and implementation of the k-best sphere decoding for mimo detection," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 491–503, 2006.

[21] P. Karamanakos, T. Geyer, and R. Kennel, "Suboptimal search strategies with bounded computational complexity to solve long-horizon direct model predictive control problems," in *2015 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2015, pp. 334–341.

[22] E. Zafra, S. Vazquez, A. Marquez Alcaide, L. G. Franquelo, J. I. Leon, and E. Perez Martin, "K-best sphere decoding algorithm for long prediction horizon fcs-mpc," *IEEE Trans. Ind. Electron.*, vol. 69, no. 8, pp. 7571–7581, 2022.

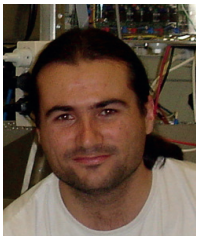
[23] J. H. Silverman, *An Introduction to Lattices, Lattice Reduction, and Lattice-Based Cryptography*. IAS Park City Mathematics Series, 2010.

[24] A. Cline, C. Moler, G. Stewart, and J. Wilkinson, "An estimate for the condition number of a matrix," *SIAM Journal on Numerical Analysis*, vol. 16, no. 2, pp. 368–375, 1979.

[25] J. Proakis and M. Salehi, *Digital Communications*, McGrow-Hill, Ed., 1995.

Eduardo Zafra was born in Seville, Spain in 1994. He received the M.S. and PhD degrees in automatics, electronics and telecommunications engineering from the University of Seville (US) in 2019 and 2024, respectively.

After his PhD, he joined EKS Energy, a Hitachi Group company, as R&D Engineer in the Product Development and Innovation division. His main interests include the control and modulation of power converters and drives, with special focus on renewable energy and energy storage applications, and digital design for embedded systems.



Sergio Vazquez (Fellow, IEEE) was born in Seville, Spain, in 1974. He received the M.S. and PhD degrees in industrial engineering from the University of Seville (US) in 2006, and 2010, respectively. Since 2002, he is with the Power Electronics Group working in R&D projects. He is an Associate Professor with the Department of Electronic Engineering.

His research interests include power electronics systems, modeling, modulation and control of power electronics converters applied to renewable energy technologies.

Dr. Vazquez was recipient as coauthor of the 2012 Best Paper Award of the IEEE Transactions on Industrial Electronics and, 2015 and 2022 Best Paper Award of the IEEE Industrial Electronics Magazine. He is involved in the Energy Storage Technical Committee of the IEEE industrial electronics society and is currently serving as an Associate Editor of the IEEE Transactions on Industrial Electronics.

Tobias Geyer (Fellow, IEEE, 2022) received the Dipl.-Ing. degree in electrical engineering, the Ph.D. degree in control engineering and the Habilitation degree in power electronics from ETH Zurich in the years 2000, 2005 and 2017, respectively.

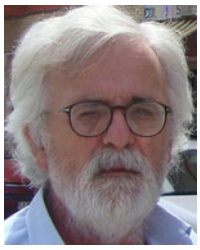
After his Ph.D., he spent three years with GE Global Research, Munich, Germany, three years with the University of Auckland, Auckland, New Zealand, and eight years with ABB's Corporate Research Centre, Baden-Dattwil, Switzerland. In 2020, he joined ABB's Medium-Voltage Drive division as R&D Platform Manager of the ACS6080. In 2022, he became a Corporate Executive Engineer. He has been an extraordinary Professor with Stellenbosch University, Stellenbosch, South Africa, since 2017.

He is the author of more than 40 patent families, 170 publications, and the book "Model predictive control of high power converters and industrial drives" (Wiley, 2016). He teaches a regular course on model predictive control at ETH Zurich. His research interests include medium-voltage and low-voltage drives, utility-scale power converters, optimized pulse patterns and model predictive control.

Dr. Geyer received the PELS Modeling and Control Technical Achievement Award in 2022, the Semikron Innovation Award in 2021, and the Nagamori Award in 2021. He was also the recipient of two Prize Paper Awards of IEEE Transactions and three Prize Paper Awards at IEEE conferences. He is a former Associate Editor of IEEE Transactions on Industry Applications (from 2011 until 2014) and IEEE Transactions on Power Electronics (from 2013 until 2019). He was an International Program Committee Vice Chair of the IFAC conference on Nonlinear Model Predictive Control in Madison, WI, USA, in 2018. He was a Distinguished Lecturer of the IEEE Power Electronics Society from 2020 to 2023. He is a Fellow of the IEEE.

Ricardo P. Aguilera (S'01–M'12) received the B.Sc. degree in electrical engineering from the Universidad de Antofagasta, Antofagasta, Chile; the M.Sc. degree in electronics engineering from the Universidad Tecnica Federico Santa Maria, Valparaíso, Chile; and the Ph.D. degree in electrical engineering from The University of Newcastle (UoN), Newcastle, NSW, Australia, in 2003, 2007, and 2012, respectively.

From 2012 to 2013, he was a Research Academic at UoN, where he was part of the Centre for Complex Dynamic Systems and Control. From 2014 to 2016, he was a Senior Research Associate at The University of New South Wales, Australia, where he was part of the Australian Energy Research Institute. Since September 2016, he has been with the School of Electrical and Data Engineering, at the University of Technology Sydney, Australia, where he currently holds an Associate Professor position. From 2021 to 2022, he was the Vice-Chair of the IEEE NSW Joint Chapter PEL/IES/IAS. Currently, he serves as an Associate Editor of the IEEE Journal of Emerging and Selected Topics in Power Electronics. His main research interests include theoretical and practical aspects on model predictive control with application to power electronics, renewable energy integration, and microgrids.



Emilio Freire was born in Reus, Spain, in 1952. He received the M.S. Ingeniero Industrial and Ph.D. Doctor Ingeniero Industrial degrees from the Universidad de Sevilla, Seville, Spain, in 1975 and 1982, respectively.

Since 1978, he has been with the Departamento de Matematica Aplicada, Universidad de Sevilla, where he is currently a Professor and the Director of the Dynamical Systems Group. From 1991 to 1998, he was also the Head of the Department. His research interests include mathematical modeling of circuits and systems, bifurcation theory, and nonlinear problems in Hamiltonian systems and other dynamical systems.

Leopoldo G. Franquelo (Life Fellow, IEEE) was born in Malaga, Spain. He received the M.Sc. and Ph.D. degrees in electrical engineering from the Universidad de Sevilla, Seville, Spain, in 1977 and 1980, respectively. He was associate professor from 1982–86 at the Electronics Engineering Department in Sevilla University and currently is professor at the Electronics Engineering Department in Sevilla University since 1986 and a 1000 Talent Professor at the Department of Control Science and Engineering in Harbin Institute of Technology since 2016.

His current research interests include modulation techniques for multilevel inverters and application to power electronic systems for renewable energy systems. He has participated in more than 100 Industrial and R&D projects and has published more than 300 papers, 76 of them in IEEE Journals.

Dr. Franquelo is an IEEE Fellow since 2005 and an IEEE Industrial Electronics Society (IES) Distinguished Lecturer since 2006. In the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, he became an Associate Editor in 2007, Co-Editor-in-Chief in 2014, and the Editor-in-Chief in 2016. He was the President of the IES (2010–2011) and is an IES AdCom Life member. In 2009 and 2013, he received the prestigious Andalusian Research Award and FAMA Award recognizing the excellence of his research career. He has received a number of Best Paper Awards from IEEE journals. In 2012 and 2015, he was the recipient of the Eugene Mittelmann Outstanding Research Achievement Award and the Anthony J. Hornfeck Service Award from IEEE-IES, respectively.