IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 2 April 2024, accepted 27 April 2024, date of publication 2 May 2024, date of current version 13 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3396352

RESEARCH ARTICLE

Linear State Signal Shaping Explicit Model Predictive Control Using Tensor Decompositions

CARLOS CATERIANO YÁÑEZ[1,2,3], GEORG PANGALOS[1], JAN-HENRIK MEYER[4], GERWALD LICHTENBERG[3], AND JAVIER SANCHIS SÁEZ[2]

[1] Application Center for Integration of Local Energy Systems ILES, Fraunhofer Institute for Wind Energy Systems IWES, 21029 Hamburg, Germany
[2] Instituto Universitario de Automática e Informática Industrial, Universitat Politècnica de València, 46022 Valencia, Spain
[3] Faculty Life Sciences, Hamburg University of Applied Sciences, 21033 Hamburg, Germany
[4] Panasonic Industrial Devices Europe, 21337 Lüneburg, Germany

Corresponding author: Carlos Cateriano Yáñez (carlos.cateriano.yanez@iwes.fraunhofer.de)

This work was supported in part by the Project Northern German Living Laboratory (NRL) by the Federal Ministry for Economic Affairs and Climate Action.

ABSTRACT Due to the increasing use of nonlinear loads in modern power systems, harmonic currents have become a more prominent problem for power quality. Typically, harmonic currents are compensated by using shunt active power filters. Recently, a novel constrained linear state signal shaping model predictive controller has been proposed for shunt active power filter control. However, due to the high computational requirements of online quadratic programming solvers, the real-time implementation of this solution is quite challenging. Therefore, the present work proposes the use of a linear state signal shaping *explicit* model predictive control formulation, such that the optimizations are done offline. However, the generated offline data introduces a large memory footprint, hindering real-time implementation. To break the curse of dimensionality, a tensor representation is proposed, which can be efficiently compressed via tensor decomposition methods. The proposed approach was tested in simulation and was able to provide good results. Due to the use of efficient tensor decomposition methods, a considerable reduction of the memory requirement could be achieved.

INDEX TERMS Harmonic compensation, explicit model predictive control, active power filter, tensor decomposition.

I. INTRODUCTION

The number of converter-connected generation units and consumers is constantly rising, which poses, among other power quality problems, the introduction of harmonic disturbance into the grid, [1]. For the reduction of harmonic disturbance, several approaches are available, shunt active power filters (SAPFs) being the most prominent one, with an active field of research, e.g., [2] uses Lyapunov theory and [3] neural networks.

Recently, a new linear state signal shaping model predictive control (LS^{3}MPC) approach was proposed by [4], which is based on model predictive control (MPC) with constraints, i.e., a quadratic program (QP). When dealing with harmonic compensation, the LS^{3}MPC can use a harmonic shape class as reference dynamics for its compensation, as seen in [4] (for a comparison of the base concept against classical approaches see [5]). In general, MPC offers many advantages over classical control approaches, like look-ahead action and built-in constraint handling. However, despite recent advancements in optimization solvers, see [6] and [7], the online computation of an MPC is still quite demanding for real-time applications, particularly at small timescales like in harmonic compensation. Thus, a reasonable approach is to use an explicit MPC formulation, which enables the offline precomputation of the control laws.

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed F. Zobaa.

In explicit MPC approaches, the control problem is formulated as a multi-parametric quadratic program (mp-QP), which solution leads to a piecewise affine (PWA) function containing the control law over each polytopic region of a set of convex critical regions [8]. The explicit MPC implementation is divided in two phases. The first phase is offline, where the PWA control laws are calculated for each critical region and stored. The second phase is online, where the real-time controller solves a point location problem to identify the currently active critical region and its corresponding control law, which is easily evaluated, [9].

However, depending on the original problem dimensions, the memory footprint of the mapped solution regions and the run time effort to identify the critical region can rapidly scale up. Therefore, different approaches have been explored in literature to reduce the explicit MPC problem complexity. Many approaches focus on the offline phase, simplifying the PWA solution by, e.g., finding its minimal polyhedral region representation [10], or looking for equivalent but more efficient formulations like its lattice representation [11]. More recently in [12], a region-free approach is proposed, which further reduces the memory and computation burden of the offline phase, as no critical regions but only their optimality condition variables need to be constructed or stored [13]. These approaches are also known as combinatorial or implicit enumeration techniques, which often order the active set candidates in a combinatorial tree, see [14], for more recent advancements.

Alternatively, other approaches focus on the online phase, trying to speed up the point location problem to reduce the worst-case sample time like [15], which proposes a hash-based approach in contrast to classic direct search. Some approaches like [16], which uses binary search trees instead in the lattice representation of the PWA, offer trade-offs between preprocessing time, storage requirement, and online computation time. Other techniques, like constraint sorting [17], also help accelerate the online evaluation by region discarding.

Nevertheless, despite the improvements achieved by the aforementioned approaches, some MPC setups are still too complex. This is particularly the case when long prediction horizons are required, as it increases the region partitions, thus hindering its real-time feasibility. In this context, aiming for simplified suboptimal solutions that approximates the explicit MPC problem can be considered as an option if the application allows it. Approximation approaches have been considered since the early stages of the explicit MPC, see [18]. Many approaches aim at approximating the PWA solution with simpler objects, like orthogonal hypercubes [19] or simplices [20]. For a more comprehensive review of further explicit MPC approaches refer to [21]. While many of these approaches greatly reduce the explicit MPC complexity and memory burden, they are still susceptible to the curse of dimensionality, as the number of combinations keeps growing, even well-built search trees become unmanageable.

The aim of this publication is to find an explicit MPC formulation that is suitable for the LS^{3}MPC approach in the application field of SAPF for harmonic compensation. In this context, due to the large prediction horizon requirements of the approach [4] and the flexible margins of the application field, approximate approaches are a reasonable course of action.

The proposed explicit LS^{3}MPC approach aims to exploit the strength of tensors, in particular their decomposition in low-rank approximations, to achieve a more efficient solution region representation for the explicit MPC. Tensor decomposition is already extensively used in the fields of signal processing, statistics, data mining, machine learning, and many more, e.g., [22]. Tensors can also be used in the parameter space, e.g., for describing multilinear models [23], because of their close structural relation. For explicit MPC, tensors have been sparsely used, e.g., the tensor product in the multiresolution approximation approach in [24]. However, to the best of the authors' knowledge, tensor decomposition methods remain untapped.

Similarly to other approximation approaches, the explicit LS^{3}MPC uses simpler orthotope objects to approximate the PWA function. However, in contrast to other methods, each dimension has a constant partition, leading to an equidistant mesh that allows for a tensor formulation that approximates the PWA function. These partition restrictions essentially make the point localization problem negligible, as each dimension is uniformly sampled, greatly simplifying the online phase. Nevertheless, to attain an acceptable approximation accuracy, a really fine mesh is required, which consequentially leads to a prohibitive number of partitions to be stored in the offline phase. This is when the tensor decomposition methods, enabled by the tensor formulation, come to play, alleviating the curse of dimensionality of the offline phase significantly via low-rank approximations. Moreover, due to the nature of the application field and the LS^{3}MPC formulation, an additional frequency domain approximation is used to further reduce the size of the problem and its dependence on the prediction horizon (in terms of dimension, not combinations).

The proposed explicit LS^{3}MPC presents a promising approach to tackle the curse of dimensionality. Nevertheless, it still has open problems, particularly in the offline calculation of the full tensor and its decomposition. In this context, this manuscript aims to determine whether a low-rank tensor decomposition approximation can satisfactorily capture the dynamics of an explicit MPC, in particular for the LS^{3}MPC setup.

The paper is organized as follows. The LS^{3}MPC is introduced in section II. The explicit LS^{3}MPC with the notation in tensor format, as well as the decomposition of the resulting tensor, is given in section III. The application example is described, and simulation results are given in section IV. Finally, conclusions are drawn in section V.

II. LINEAR STATE SIGNAL SHAPING MODEL PREDICTIVE CONTROL

This section introduces the base MPC and linear state signal shaping (LS^3) theory required to formulate the LS^{3}MPC problem.

A. MODEL PREDICTIVE CONTROL

The MPC framework used for this work considers a discrete-time linear state space system formulation for the plant, which is given as

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{D}\mathbf{d}(k), \tag{1}$$

with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^l$, and $\mathbf{d} \in \mathbb{R}^d$ as state, input, and disturbance vectors; with $\{n, l, d\} \in \mathbb{Z}_{\geq 0}$ as number of states, inputs, and disturbances respectively; and $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times l}$, and $\mathbf{D} \in \mathbb{R}^{n\times d}$ as system, input, and disturbance matrices, for discrete time $k \in \mathbb{Z}_{\geq 0}$. For a given finite prediction horizon H_p, the MPC control action $\mathbf{u}$ for stabilizing (1) starting at $\mathbf{x}(0) = \mathbf{x}_0$, is obtained by solving the following QP

$$\min_{\mathbf{u}} \sum_{k=0}^{H_p-1} \mathbf{x}^\top(k)\mathbf{Q}(k)\mathbf{x}(k) + \mathbf{u}^\top(k)\mathbf{R}(k)\mathbf{u}(k), \tag{2}$$

$$\text{s.t. } \mathbf{x}_{\min} \leq \mathbf{x}(k) \leq \mathbf{x}_{\max}\ \forall\, k \in \left\{1, \dots, H_p\right\}, \tag{2a}$$

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{D}\mathbf{d}(k)$$
$$\forall k \in \left\{0, \dots, H_p - 1\right\}, \tag{2b}$$

with $\mathbf{Q}(k) \in \mathbb{R}^{n\times n}$ and $\mathbf{R}(k) \in \mathbb{R}^{l\times l}$ standing for the block diagonal elements of their corresponding state and input weighting matrices

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}(0) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}(1) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{Q}(H_p-1) \end{pmatrix} \in \mathbb{R}^{nH_p\times nH_p} \succeq \mathbf{0}, \tag{2c}$$

$$\mathbf{R} = \begin{pmatrix} \mathbf{R}(0) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{R}(1) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{R}(H_p-1) \end{pmatrix} \in \mathbb{R}^{lH_p\times lH_p} \succ \mathbf{0}, \tag{2d}$$

where $\mathbf{0}$ is a matrix of zeros of appropriate size, and $\mathbf{x}_{\min} \in \mathbb{R}^n$ and $\mathbf{x}_{\max} \in \mathbb{R}^n$ stand for the lower and upper bounds of the states vector $\mathbf{x}$, [25]. Naturally, more formulations and multiple input and state constraint combinations are possible.

Parting from (1), lifted system equations for the whole prediction horizon H_p can be formulated as

$$\mathcal{X}(k) = \Psi\mathbf{x}(k) + \Phi\,\mathcal{U}(k) + \Gamma\,\mathcal{D}(k), \tag{3}$$

where

$$\mathcal{X}(k) = \left(\mathbf{x}^\top(k+1)\ \mathbf{x}^\top(k+2)\ \dots\ \mathbf{x}^\top\left(k+H_p\right)\right)^\top \in \mathbb{R}^{nH_p}, \tag{3a}$$

$$\Psi = \left(\mathbf{A}^\top \quad \mathbf{A}^{2\top} \quad \dots \quad \mathbf{A}^{H_p\top}\right)^\top \in \mathbb{R}^{nH_p\times n}, \tag{3b}$$

$$\Phi = \begin{pmatrix} \mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{AB} & \mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{H_p-1}\mathbf{B} & \mathbf{A}^{H_p-2}\mathbf{B} & \dots & \mathbf{B} \end{pmatrix} \in \mathbb{R}^{nH_p\times lH_p}, \tag{3c}$$

$$\mathcal{U}(k) = \left(\mathbf{u}^\top(k) \quad \mathbf{u}^\top(k+1) \quad \dots \quad \mathbf{u}^\top\left(k+H_p-1\right)\right)^\top \in \mathbb{R}^{lH_p}, \tag{3d}$$

$$\Gamma = \begin{pmatrix} \mathbf{D} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{AD} & \mathbf{D} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{H_p-1}\mathbf{D} & \mathbf{A}^{H_p-2}\mathbf{D} & \dots & \mathbf{D} \end{pmatrix} \in \mathbb{R}^{nH_p\times dH_p}, \tag{3e}$$

$$\mathcal{D}(k) = \left(\mathbf{d}^\top(k) \quad \mathbf{d}^\top(k+1) \quad \dots \quad \mathbf{d}^\top\left(k+H_p-1\right)\right)^\top \in \mathbb{R}^{dH_p}. \tag{3f}$$

The previous equations can be used to reformulate (2), such that it can be solved with standard QP solvers, e.g., operator splitting quadratic program (OSQP) solver by [26], leading to

$$V(\mathbf{z}(k)) = \min_{\mathcal{U}(k)} \frac{1}{2}\mathcal{U}^\top(k)\mathbf{H}\,\mathcal{U}(k) + \mathbf{z}^\top(k)\mathbf{F}\,\mathcal{U}(k), \tag{4}$$

$$\text{subject to } \mathbf{G}\,\mathcal{U}(k) \leq \mathbf{W} + \mathbf{E}\mathbf{z}(k), \tag{4a}$$

where

$$\mathbf{z}(k) = \left(\mathbf{x}^\top(k)\ \mathcal{D}^\top(k)\right)^\top \in \mathbb{R}^{n+dH_p}, \tag{4b}$$

$$\mathbf{H} = 2\left(\Phi^\top\mathbf{Q}\Phi + \mathbf{R}\right) \in \mathbb{R}^{lH_p\times lH_p}, \tag{4c}$$

$$\mathbf{F} = 2\begin{pmatrix} \Psi^\top\mathbf{Q}\Phi \\ \Gamma^\top\mathbf{Q}\Phi \end{pmatrix} \in \mathbb{R}^{(n+dH_p)\times lH_p}, \tag{4d}$$

$$\mathbf{G} = \left(\Phi^\top \quad -\Phi^\top\right)^\top \in \mathbb{R}^{2nH_p\times lH_p}, \tag{4e}$$

$$\mathbf{W} = \left(\mathbf{x}_{\max}^\top \otimes \mathbf{1}_{H_p}^\top \quad \mathbf{x}_{\min}^\top \otimes \mathbf{1}_{H_p}^\top\right)^\top \in \mathbb{R}^{2nH_p}, \tag{4f}$$

$$\mathbf{E} = \begin{pmatrix} -\Psi & -\Gamma \\ \Psi & \Gamma \end{pmatrix} \in \mathbb{R}^{2nH_p\times(n+dH_p)}, \tag{4g}$$

with $\mathbf{1}_{H_p}$ as a column vector of ones of size H_p, [8].

Remark 1 (Convexity): Note that to ensure that (4) is a convex problem, the Hessian $\mathbf{H}$ in (4c), needs to be symmetric and positive semidefinite [27], i.e., $\mathbf{H} = \mathbf{H}^\top \succeq \mathbf{0}$. This is ensured by the choice of $\mathbf{Q}$ and $\mathbf{R}$ in (2).

B. LINEAR STATE SIGNAL SHAPING

The LS^3 theory centers around shaping linear system states, like $\mathbf{x}$ from (1), into "shapes" belonging to specific linear dynamics, captured by a so-called "shape class", as defined in the following.

Definition 1 (Shape Class): Given a discrete-time state space system with state vector $\mathbf{x} \in \mathbb{R}^n$, starting at discrete time k, let

$$\mathcal{X}_L(k) = \left(\mathbf{x}^\top(k) \;\; \mathbf{x}^\top(k+1) \;\; \dots \;\; \mathbf{x}^\top(k+L-1)\right)^\top \in \mathbb{R}^{nL}, \tag{5a}$$

$$\mathcal{X}_T(k) = \left(\mathbf{x}^\top(k) \;\; \mathbf{x}^\top(k+1) \;\; \dots \;\; \mathbf{x}^\top(k+T-1)\right)^\top \in \mathbb{R}^{nT}, \tag{5b}$$

where $L \in \mathbb{Z}_{\geq 1}$ and $T \in \mathbb{Z}_{\geq 1}, L \geq T$, then, the set

$$\hat{\mathscr{X}}_{\mathbf{V}} = \{\mathcal{X}_L(k)\,|\mathbf{V}\mathcal{X}_T(j) = \mathbf{0}\,\forall j = k, \dots, k+L-T-1\}, \tag{5c}$$

contains any sequence $\mathcal{X}_L(k)$ consisting of L consecutive discrete-time $\mathbf{x}$ instances, which belong to the shape class given by the shape matrix $\mathbf{V} \in \mathbb{R}^{s \times nT}$, which defines s linear difference equations for a sequence of T consecutive discrete-time $\mathbf{x}$ instances [28].

Definition 1 outlines the kernel of the shape class linear map $\ker(\mathbf{V})$, which consists of a set of linear difference equations that define the desired shape dynamics.

Remark 2 (Shape Class Set): Note that the shape class definition set $\hat{\mathscr{X}}_{\mathbf{V}}$ in (5c) is a more restrictive definition than the general shape class set $\mathscr{X}_{\mathbf{V}}$ extending to infinity, as given by [28]. However, due to the finite nature of the MPC prediction horizon H_p, this finite formulation subset is proposed here.

The control goal for LS^3 is to find solutions that lead the system to state sequences $\mathcal{X}_L$ that piecewise, i.e., $\mathcal{X}_T$, lie within the kernel of a given shape matrix or at least within its vicinity. When a solution state sequence does not belong to the kernel exactly, i.e., $\mathbf{V}\mathcal{X}_T(k) \neq \mathbf{0}$, it defines a shape residual [28]

$$\|\mathbf{V}\mathcal{X}_T(k)\|_2\,. \tag{6}$$

The core idea of the LS^{3}MPC is to find an optimal state sequence $\mathcal{X}_L(k)$ that minimizes the shape residual in (6) throughout its prediction horizon, i.e., $\mathcal{X}_L(k) = \mathcal{X}(k)$, since $L = H_p$. This can be achieved by formulating the following minimization problem

$$\min_{\mathcal{X}(k)} \mathcal{X}^\top(k)\mathbf{P}_{\mathbf{V}}^\top\mathbf{P}_{\mathbf{V}}\mathcal{X}(k), \tag{7}$$

with the pattern band matrix

$$\mathbf{P}_{\mathbf{V}} = \begin{pmatrix} \mathbf{V}_1 & \mathbf{V}_2 & \cdots & \mathbf{V}_T & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_1 & \mathbf{V}_2 & \cdots & \mathbf{V}_T & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{V}_1 & \mathbf{V}_2 & \cdots & \mathbf{V}_T \end{pmatrix} \in \mathbb{R}^{s(H_p-T+1) \times nH_p}, \tag{7a}$$

where

$$\mathbf{V}_j = \mathbf{V}\begin{pmatrix} \mathbf{0}_{n(j-1)\times n} \\ \mathbf{I}_{n\times n} \\ \mathbf{0}_{n(T-j)\times n} \end{pmatrix} \in \mathbb{R}^{s\times n}, \tag{7b}$$

are block matrices from the shape class matrix $\mathbf{V}$ [28], with zero matrix $\mathbf{0}_{i\times j} \in \mathbb{Z}^{i\times j}$ and identity matrix $\mathbf{I}_{i\times i} \in \mathbb{Z}^{i\times i}$. As shown by [4], this minimization problem can easily be embedded into (4) QP formulation by setting the state weighting cost matrix as $\mathbf{Q} = \mathbf{P}_{\mathbf{V}}^\top\mathbf{P}_{\mathbf{V}}$. This approach directly affects the Hessian in (4c), which becomes non-diagonal, but still needs to be positive semidefinite to ensure (4) solvability, see Remark 1. For further remarks on the closed-loop behavior of the LS^{3}MPC and its design parameters, please refer to [4] and [28].

III. LINEAR STATE SIGNAL SHAPING EXPLICIT MODEL PREDICTIVE CONTROL IN TENSOR FORM

This section recaps the explicit MPC theory and introduces the explicit LS^{3}MPC. The section concludes with a tensor-based approach to reduce the storage demand of the explicit LS^{3}MPC.

A. EXPLICIT MODEL PREDICTIVE CONTROL

One of the main objectives of an explicit MPC is to move most of the computation effort offline [8]. The key idea is to treat (4) as an mp-QP with an *explicit* dependency on the vector of parameters $\mathbf{z}(k)$ as given in (4b). Then, the mp-QP problem solution $\mathcal{U}^\star(k)$ can be computed offline, leading to a piecewise affine function of $\mathbf{z}(k)$ [8]. Such piecewise affine function is given as follows

$$\mathbf{w}(\mathbf{z}(k)) = \mathbf{M}^i\mathbf{z}(k) + \mathbf{c}^i \quad \forall\, \mathbf{z} \in \mathbf{CR}_i, \tag{8}$$

where $\mathbf{M}^i \in \mathbb{R}^{lH_p \times n+dH_p}$ is the slope matrix and $\mathbf{c}^i \in \mathbb{R}^{lH_p}$ is the offset vector for the convex polyhedral region $\mathbf{CR}_i$, which together conform a subset for $\mathbf{z} \in \mathscr{Z} \subseteq \mathbb{R}^{n+dH_p}$.

The first step to transform (4) into an mp-QP is to perform the change of variable

$$\mathbf{w} = \mathcal{U}(k) + \mathbf{H}^{-1}\mathbf{F}^\top\mathbf{z}(k), \tag{9}$$

leading to

$$V_{\mathbf{w}}(\mathbf{z}(k)) = \min_{\mathbf{w}} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{H}\mathbf{w}, \tag{10}$$

$$\text{subject to} \quad \mathbf{G}\mathbf{w} \leq \mathbf{W} + \mathbf{S}\mathbf{z}(k), \tag{10a}$$

where

$$\mathbf{S} = \mathbf{E} + \mathbf{G}\mathbf{H}^{-1}\mathbf{F}^\top \in \mathbb{R}^{2nH_p \times (n+dH_p)}, \tag{10b}$$

$$V_{\mathbf{w}}(\mathbf{z}(k)) = V(\mathbf{z}(k)) + \frac{1}{2}\mathbf{z}^\top(k)\mathbf{F}\mathbf{H}^{-1}\mathbf{F}^\top\mathbf{z}(k). \tag{10c}$$

The solution to the mp-QP in (10), as detailed in [8], is the following

$$\mathbf{w} = \tilde{\mathbf{M}}\left(\tilde{\mathbf{W}} + \tilde{\mathbf{S}}\mathbf{z}(k)\right), \tag{11}$$

where $\tilde{\mathbf{M}} = \mathbf{H}^{-1}\tilde{\mathbf{G}}^\top\left(\tilde{\mathbf{G}}\mathbf{H}^{-1}\tilde{\mathbf{G}}^\top\right)^{-1}$, and $\tilde{\mathbf{W}}$, $\tilde{\mathbf{S}}$, and full rank $\tilde{\mathbf{G}}$ are the set of active constraints, i.e.,

$$\tilde{\mathbf{G}}\mathbf{w} - \tilde{\mathbf{W}} - \tilde{\mathbf{S}}\mathbf{z}(k) = \mathbf{0}. \tag{12}$$

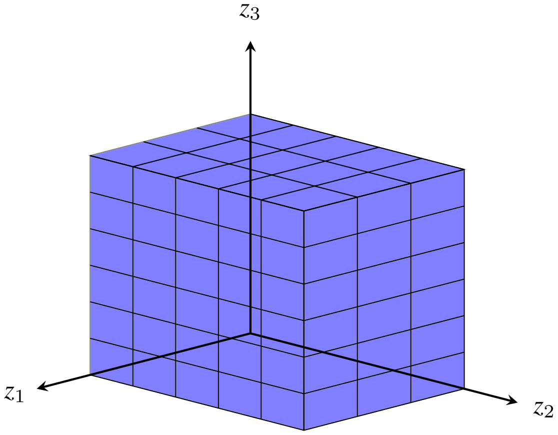

FIGURE 1. Meshing of a 3-D state space.

Replacing (11) in (9) leads to

$$\mathcal{U}(k, \mathbf{z}) = \left(\tilde{\mathbf{M}}\tilde{\mathbf{S}} - \mathbf{H}^{-1}\mathbf{F}^{\top}\right)\mathbf{z}(k) + \tilde{\mathbf{M}}\tilde{\mathbf{W}}. \tag{13}$$

These matrix function parameters are stored as follows

$$\mathbf{K} = \left(\tilde{\mathbf{M}}\tilde{\mathbf{S}} - \mathbf{H}^{-1}\mathbf{F}^{\top} \;\; \tilde{\mathbf{M}}\tilde{\mathbf{W}}\right) \in \mathbb{R}^{lH_p \times (dH_p+n+1)}, \tag{14}$$

for the corresponding slope and offset. To fulfill (10a) considering (11), it follows

$$\mathbf{G}\tilde{\mathbf{M}}\left(\tilde{\mathbf{W}} + \tilde{\mathbf{S}}\mathbf{z}(k)\right) \leq \mathbf{W} + \mathbf{S}\mathbf{z}(k). \tag{15}$$

Moreover, the Lagrange multipliers for the first-order Karush-Kuhn-Tucker (KKT) optimality conditions for (10) must remain non-negative

$$\tilde{\lambda} = -\left(\tilde{\mathbf{G}}\mathbf{H}^{-1}\tilde{\mathbf{G}}^{\top}\right)^{-1}\left(\tilde{\mathbf{W}} + \tilde{\mathbf{S}}\mathbf{z}(k)\right) \in \mathbb{R}^{2nH_p} \geq \mathbf{0}. \tag{16}$$

Considering (15) and (16) and removing the redundant constraints in (10), as detailed in [8], a compact representation for the critical region $\mathbf{CR}_0$ for a specific $\mathbf{z}_0$ can be obtained. Next, the remaining region $\mathscr{Z} - \mathbf{CR}_0$ needs to be explored, repeating the same procedure to find all new critical regions. The steps for this procedure can be summarized as follows:

1) solve mp-QP in (10) for a chosen $\mathbf{z}_i$,
2) find the set of active constraints (12),
3) calculate the corresponding matrix function parameters $\mathbf{K}_i$ as given in (14),
4) compute a compact representation of the region $\mathbf{CR}_i$ considering (15) and (16),
5) choose a new $\mathbf{z}_i$ and repeat.

B. EXPLICIT LS3MPC

This subsection starts by introducing the meshing approach, followed by a frequency domain parameterization method for the variables in $\mathbf{z}$.

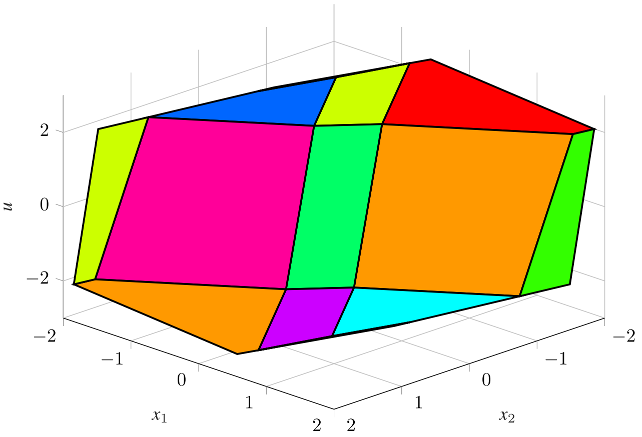

FIGURE 2. Piecewise linear function with polyhedral regions.

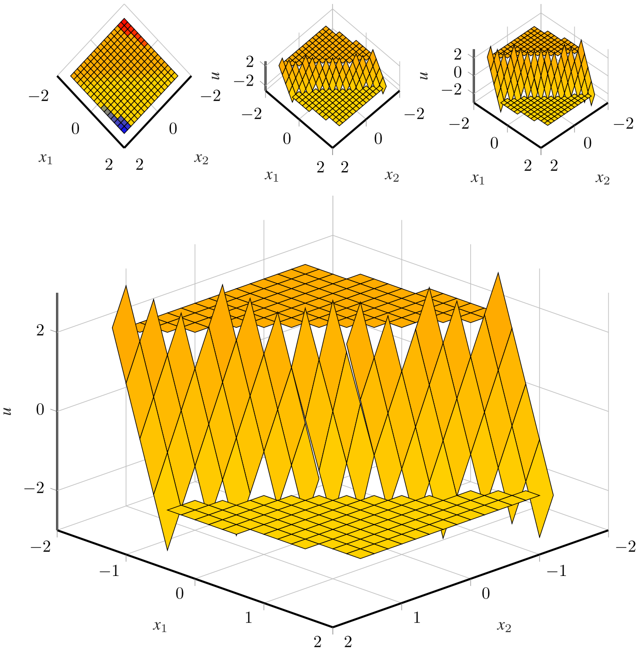

FIGURE 3. Piecewise linear function of the proposed meshing approach from different perspectives.

1) MESHING APPROACH

In contrast to the usual approach of defining polyhedral regions $\mathbf{CR}_i$ in the state space, this publication uses the meshing approach introduced in [29] to reduce the number of variables to be stored. In this approach, the storage demand for the regions is reduced because of the mesh structure. Furthermore, the disturbance and the control input are parametrized in the frequency domain. Both will be described in the following.

The state space is meshed in all variables leading to orthotopes, as displayed in Fig. 1 for the case of three states.

For each orthotope $j = 1, \ldots, \Pi_{i=1}^{z} n_i$, the function parameters matrix $\mathbf{K}_j \in \mathbb{R}^{lH_p \times (dH_p+n+1)}$, as given in (14), is calculated and stored in a tensor

$$\mathsf{T} \in \mathbb{R}^{n_1 \times \ldots \times n_z \times lH_p \times (dH_p+n+1)}, \tag{17}$$

where n_i is the number of orthotopes in the direction of state x_i, for $i = 1, \ldots, n_z$. The parameters $\mathbf{z}_j$ for calculating the matrix $\mathbf{K}_j$ correspond to the centroid of the respective orthotope region. Because the dimensions of the tensor T and their sizes implicitly give the mesh structure, no additional information needs to be stored about it, unlike polyhedral regions, which ultimately trivializes the point location problem. Next, a minimal example is introduced to help visualize the proposed meshing approach.

Example 1 (Piecewise Linear Function): The optimization problem (4) was solved for the state-space model

$$\mathbf{x}(k+1) = \begin{pmatrix} 0.7326 & -0.0861 \\ 0.1722 & 0.9909 \end{pmatrix} \mathbf{x}(k) + \begin{pmatrix} 0.0609 \\ 0.0064 \end{pmatrix} u(k), \tag{18}$$

with sampling time $t_s = 0.1$ s and weighting matrices

$$\mathbf{Q} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \tag{19}$$

$$\mathbf{R} = 0.001. \tag{20}$$

The controller aims to drive the states to the equilibrium at the origin, i.e., $x_1 = x_2 = 0$. The prediction horizon was kept at $H_p = 2$ to keep the scale of the example to a minimum. The input was constrained to ± 2 and the states to ± 1.5. The mesh size was chosen to be $\Delta x = 0.2$ for both states.

Fig. 2 shows the standard polyhedral regions approach for explicit MPC. In contrast, Fig. 3 shows the piecewise linear function of the proposed meshing approach, which is not as smooth. At the borders of the orthotopes, discontinuities arise, which can violate the inequality constraints.

The bigger the mesh size, the more likely the inequality constraints will be violated. Naturally, the simpler mesh structure provided by the orthotopes does not match the polyhedral regions exactly, bringing a trade-off between accuracy and mesh size.

2) FREQUENCY DOMAIN PARAMETERIZATION

As stated before, the dimension of matrices $\mathbf{K}_j$ in (14) depends on the prediction horizon H_p, the number of disturbances d, the number of states n, and the number of inputs l. However, by expressing the disturbance and the control input in terms of harmonic signals, only the amplitude and phase of each harmonic need to be stored, thus removing the problem size dependency on the prediction horizon. This can be done using a discrete Fourier transform (DFT)

$$\mathcal{D}(k) \approx \mathbf{W}_\Gamma \hat{\mathcal{D}}(\omega_k), \tag{21}$$

$$\mathcal{U}(k) \approx \mathbf{W}_u \hat{\mathcal{U}}(\omega_k), \tag{22}$$

with $\omega_k = 2\pi f k$ for frequency f, and Fourier matrices $\mathbf{W}_\Gamma \in \mathbb{C}^{dH_p \times 2dn_d}$ and $\mathbf{W}_u \in \mathbb{C}^{lH_p \times 2ln_u}$, where n_u is the number of harmonics considered for the control input DFT and n_d is the number of harmonics considered for the disturbance DFT. The elements of each Fourier matrix are given as

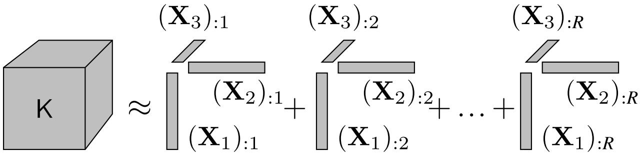

FIGURE 4. **3D rank *R* CP tensor.**

$$\mathbf{W}_{\hat{n},k} = \frac{1}{N} \cos\left(\frac{2\pi}{N} k\hat{n}\right) + \mathrm{j} \sin\left(\frac{2\pi}{N} k\hat{n}\right), \tag{23}$$

for columns k, rows $\hat{n}$, and $N = \frac{1}{ft_s}$ for sampling time t_s. This change of variable leads to a reformulation of (4) as

$$V(\mathbf{z}(k)) = \min_{\hat{\mathcal{U}}(\omega_k)} \frac{1}{2} \hat{\mathcal{U}}^\top(\omega_k) \mathbf{W}_u^\top \mathbf{H}\, \mathbf{W}_u \hat{\mathcal{U}}(\omega_k) + \mathbf{z}^\top(k) \mathbf{F} \mathbf{W}_u \hat{\mathcal{U}}(\omega_k), \tag{24}$$

$$\text{subject to } \mathbf{G} \mathbf{W}_u \hat{\mathcal{U}}(\omega_k) \leq \mathbf{W} + \mathbf{E}\mathbf{z}(k), \tag{24a}$$

where the parameter matrix $\Gamma_{\text{Fourier}} \in \mathbb{C}^{nH_p \times 2dn_d} = \Gamma \mathbf{W}_\Gamma$ is replaced.

This parameterization extends to the mp-QP formulation in (10). Thus, the dimensions of the matrices $\mathbf{K}_j$ under this parameterization are

$$\mathbf{K}_j \in \mathbb{R}^{2ln_u \times (2dn_d + n + 1)}. \tag{25}$$

Naturally, the viability of this parameterization will depend on the balance between the required harmonic order of the application versus the prediction horizon H_p, as will be seen in section IV. Even though the proposed harmonic parameterization dramatically reduces the scale of the problem, far more significant gains are yet to be exploited due to the tensor format chosen in (17), as seen in the following subsection.

C. TENSOR DECOMPOSITION

To reduce the storage demand for the explicit LS^{3}MPC, the tensor T can be approximated by tensor decomposition methods. Various tensor decomposition methods are available [30], [31], [32], [33]. Each of these has advantages, marked as (+), and disadvantages, marked as (-) with respect to their applicability (neutral points are marked as (o)).

Tensor trains are best in a predefined degree of accuracy (++), presenting good runtimes for decompositions (+) but only medium runtimes for reconstruction (o), and a large memory size (or smaller compression rates, respectively) (-). The canonical polyadic decomposition (CPD) is the format with the largest compression rate (++), having good runtimes for reconstruction of elements (+) but long runtimes for decomposition (-), and non-unique numeric results (- -). Hierarchical or tucker formats are optimal for generically structured data, which is not the case for the current application.

TABLE 1. Equivalent circuit microgrid parameters.

Line to grid		Line to SAPF		Parallel resistor	
R_1	L_1	R_2	L_2	R_3	R_4
$1\,\Omega$	$50\,\mathrm{mH}$	$50\,\mathrm{m}\Omega$	$500\,\mathrm{mH}$	$0.5\,\mathrm{k}\Omega$	$0.5\,\mathrm{k}\Omega$

For real-time implementation, we are looking for a format which allows performant online computations (reconstruction) with a minimal amount of memory. At this stage of the explicit LS^{3}MPC development, there is not too much focus on the offline time (for decomposition) nor on achieving perfect accuracy. Thus, CPD is currently the preferred choice, though further formats might be explored at later development stages.

The CPD relies on the canonical polyadic (CP) tensor format, introduced as follows.

Definition 2 (Canonical polyadic Tensor): A canonical polyadic (CP) tensor

$$\mathsf{K} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n]\, \lambda \in \mathbb{R}^{R_1 \times R_2 \times \cdots \times R_n}, \tag{26}$$

is a tensor of dimension $(R_1, \ldots, R_n)$, with elements given by the sums of the outer products of the column vectors of the so-called *factor matrices* $\mathbf{X}_i \in \mathbb{R}^{R_i \times R}$, weighted by the elements of the so-called *weighting* or *parameter vector* $\lambda \in \mathbb{R}^R$. An element of the multidimensional tensor K is given by

$$K_{jk\ldots p} = \sum_{i=1}^{R} \lambda_i\, (\mathbf{X}_1)_{ji}\ \ (\mathbf{X}_2)_{ki}\ \ \ldots (\mathbf{X}_n)_{pi}. \tag{27}$$

If no weighting vector is given, it is assumed to be a vector of ones, i.e., $\lambda = (1 \quad 1 \quad \ldots \quad 1)^\top$ [30].

For the case of a 3-dimensional tensor, this is illustrated for rank R in Fig. 4.

To help understand how the decomposition scales, a minimal example is given in Fig. 5, with factor matrices

$$\mathsf{K} = \left[\begin{pmatrix} 0 & 1 \\ 2 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 3 \\ 1 & -3 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right]. \tag{28}$$

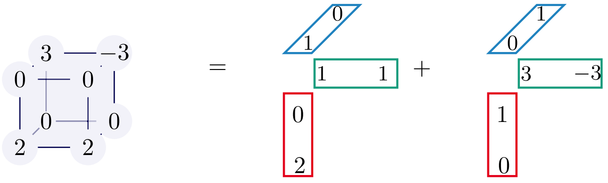

FIGURE 5. CP tensor decomposition example.

The number of columns of the factor matrices corresponds to the rank of the decomposition $R = 2$. Each dimension in the example has a size of 2, which leads to a total of 8 elements in the original tensor to be stored vs. 12 elements of the factor matrices. However, as the number of dimensions keeps increasing, the number of elements of the original tensor scales exponentially, while it scales only linearly for the decomposed version, i.e, 2^D against $2DR$, where D is the number of dimensions. Thus, by finding an adequate rank R decomposition, the curse of dimensionality is broken.

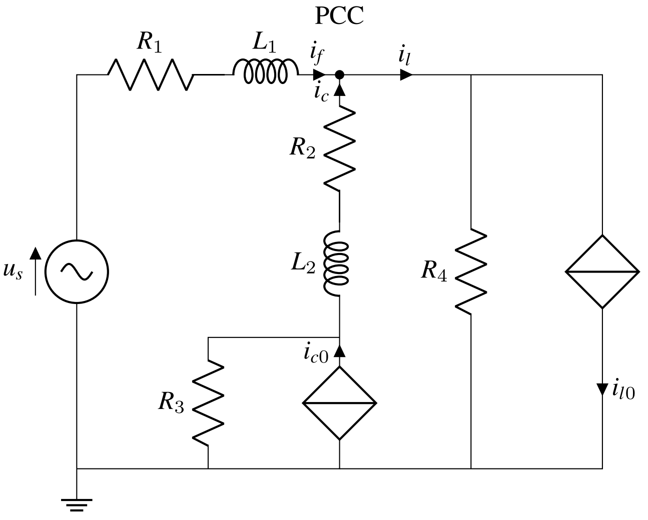

FIGURE 6. Equivalent circuit representing a 3-node microgrid, with a grid supply connection, an SAPF, and a nonlinear load.

Finding an exact decomposition with a minimal rank is nontrivial NP-hard problem. However, several toolboxes are available to find low rank approximations of full tensors. In this work, the tensorlab toolbox [34] was used.

To wrap up this section, a brief recap of the steps needed to reach this point is given in the following,

1) calculate mp-QP parameters in (4), i.e., **H**, **F**, **G**, **W**, **E**,
2) define the mesh grid size Δx for T,
3) calculate the corresponding matrix function parameters $\mathbf{K}_j$ as given in (25) for each orthotope region j in T as given in (17),
4) perform a CP tensor decomposition on T for a chosen rank R.

IV. APPLICATION EXAMPLE

This section gives a harmonic compensation example for a simple power system. The LS^{3}MPC uses a harmonic shape class to act as a reference for the compensation action of an SAPF. The objective of this section is to showcase the scalability of the approach for different low-rank tensor decomposition approximations. A more exhaustive comparison of the base concept around the harmonic shape class LS^{3}MPC against classical approaches is covered in [5].

A. MICROGRID MODELING

A simple 3-node microgrid is depicted in Fig. 6. The grid supply node is represented by the ideal voltage source u_s with feeder line resistance R_1, feeder line impedance L_1, and feeder line current i_f. The nonlinear load node is given by the resistance R_4 in parallel with the ideal current source i_{l0}, and load current i_l, which is the source of the harmonic disturbance. Finally, connected in shunt at the point of common coupling (PCC), the compensation node is depicted by the resistance R_3 in parallel with the ideal current source i_{c0}, which represent an SAPF, with line resistance R_2 and inductance L_2, and compensation current i_c. The parameters of the system are given in Table 1.

Using Kirchhoff's voltage and current laws, the following equations can be derived

$$0 = -u_s + R_1 i_f + L_1 \frac{\mathrm{d}i_f}{\mathrm{d}t} - R_2 i_c - L_2 \frac{\mathrm{d}i_c}{\mathrm{d}t} + R_3(i_{c0} - i_c), \tag{29a}$$

$$0 = -R_3(i_{c0} - i_c) + R_2 i_c + L_2 \frac{\mathrm{d}i_c}{\mathrm{d}t} + R_4(i_l - i_{l0}), \tag{29b}$$

$$i_l = i_f + i_c, \tag{29c}$$

which can be rewritten in state-space form as

$$\dot{\mathbf{x}}(t) = \begin{pmatrix} -\dfrac{R_1 + R_4}{L_1} & -\dfrac{R_4}{L_1} \\ -\dfrac{R_4}{L_2} & -\dfrac{R_2 + R_3 + R_4}{L_2} \end{pmatrix} \mathbf{x}(t) + \begin{pmatrix} 0 \\ \dfrac{R_3}{L_2} \end{pmatrix} u(t) + \begin{pmatrix} \dfrac{1}{L_1} & \dfrac{R_4}{L_1} \\ 0 & \dfrac{R_4}{L_2} \end{pmatrix} \mathbf{d}_m(t), \tag{30}$$

with state vector $\mathbf{x}(t) = \begin{pmatrix} i_f & i_c \end{pmatrix}^\top$, control input $u(t) = i_{c0}$ and the measured input disturbance $\mathbf{d}_m(t) = \begin{pmatrix} u_s & i_{l0} \end{pmatrix}^\top$.

The goal is to control the SAPF via i_{c0}, such that the harmonics injected by the nonlinear load i_{l0} are canceled, and the feeder line current i_f is free of harmonic disturbance. Since the MPC implementation used in this work is in discrete time, the system in (30) is discretized with sampling time $t_s = 40\,\mu\text{s}$, leading to a discrete-time state-space model as given in (1).

B. HARMONIC SHAPE CLASS CONTROL DESIGN

One of the aims of an SAPF is to compensate the harmonic distortion in a system, i.e., to ensure that it follows the dynamics of a fundamental harmonic. As introduced in [28], such dynamics are captured by the solution of the autonomous ordinary differential equation

$$\frac{\mathrm{d}^2 x(t)}{\mathrm{d}t^2} + (2\pi f)^2 x(t) = 0, \tag{31}$$

where $f = 50\,\text{Hz}$ denotes the frequency of the fundamental harmonic signal x. To embed these dynamics onto a shape class, (31) is discretized using central difference approximation as

$$\ddot{x}(k) \approx \frac{x(k-1) - 2x(k) + x(k+1)}{t_s^2}, \tag{32}$$

with an accuracy of order $O(t_s)^2$, [35], leading to

$$\mathbf{v} \begin{pmatrix} x(k-1) \\ x(k) \\ x(k+1) \end{pmatrix} = 0, \tag{33}$$

where $\mathbf{v}$ is the harmonic shape class vector given as

$$\mathbf{v} = \frac{1}{t_s^2} \begin{pmatrix} 1 & -2 + 4\pi^2 f^2 t_s^2 & 1 \end{pmatrix} \in \mathbb{R}^{1\times 3}. \tag{34}$$

Using the harmonic shape class vector $\mathbf{v}$ from (34) as a base, the pattern band matrix $\mathbf{P_V}$ in (8) can be built targeting the state i_f, so that the mp-QP problem in (10) can be formulated with the special state weighting matrix $\mathbf{Q} = \mathbf{P_V^\top P_V}$ as explained at the end of subsubsection II-B.

The LS^{3}MPC closed-loop block diagram is shown in Fig. 7, which corresponds to the QP in (24) with cost function short form $J(\hat{\mathcal{U}}(\omega_k), \mathbf{z}(k))$. For simplicity, direct access to the states $\mathbf{x}(k)$ is assumed, hence no observer, and a simple predictor is considered for $\mathbf{d}(k)$, which is assumed to be periodic.

C. SIMULATION SETUP

The microgrid is designed to operate with a sinusoidal supply voltage u_s of $\sqrt{2} \cdot 230\,\text{V}$ of amplitude with a fundamental frequency of $f = 50\,\text{Hz}$. Considering the sampling time $t_s = 40\,\mu\text{s}$ from the system as given in subsubsection IV-A, a prediction horizon of $H_p = 500$ is required to match the duration of one fundamental period at $f = 50\,\text{Hz}$.

Fig. 8 shows a reference LS^{3}MPC simulation operating for 4 fundamental periods, using the input sequence results of one QP solved with OSQP for each period. For this run, the target feeder line current (solid blue line) in Fig. 8 (a), was constrained to ±8 A (red dashed line) and the input weighting matrix $\mathbf{R}$ was set to be diagonal with all entries set to 1×10^{-8}. For the nonlinear load, the harmonic profile of a typical rectifier was chosen as

$$\begin{aligned} i_{l0} = {} & 9.47 \sin(\omega_f t) + 7.7 \sin(3\omega_f t + 3.2) \\ & + 4.8 \sin(5\omega_f t + 0.2) + 1.9 \sin(7\omega_f t + 3.36) \\ & + 0.8 \sin(11\omega_f t + 0.35) + 0.62 \sin(13\omega_f t + 3.55), \end{aligned} \tag{35}$$

with time and frequency spectrum as seen in Fig. 9. The compensating action of the LS^{3}MPC through i_{c0}, resulted on a satisfactory total harmonic distortion (THD) of 1.35 % for i_f, which is defined as given in Appendix.

Following the steps at the end of subsubsection III-C, the first step towards the proposed explicit LS^{3}MPC formulation is the calculation of the mp-QP parameters. While all the information for this step is already available, the current setup would lead to a tensor T with too many dimensions to handle on consumer hardware. Even though the decomposed tensor can achieve a reasonable size for computation, the full tensor still needs to be computed first, which can easily scale in complexity. Looking at (17), there is a clear dependency between the dimensions of T and n_z, the size of $\mathbf{z}$, i.e. for a second order system $\dim(\mathsf{T}) = 2 + 4n_d + 2$. By considering a fixed fundamental harmonic sine wave for u_s and restricting i_{l0} to 2 harmonics, the fundamental (also kept fixed with amplitude 10 A) and the third harmonic, i.e., $n_d = 2$, the dimensions of T are reduced to $\dim(\mathsf{T}) = 6$, with 2 dimensions corresponding to $\mathbf{K}$, 2 for the states initial conditions, and 2 for the third harmonic components of i_{l0}.

Continuing with the steps at the end of subsubsection III-C, assuming a state mesh size of $\Delta x = 0.5\,\text{A}$ for the range ±10 A and a mesh size of 2.5 A with range ±5 A for

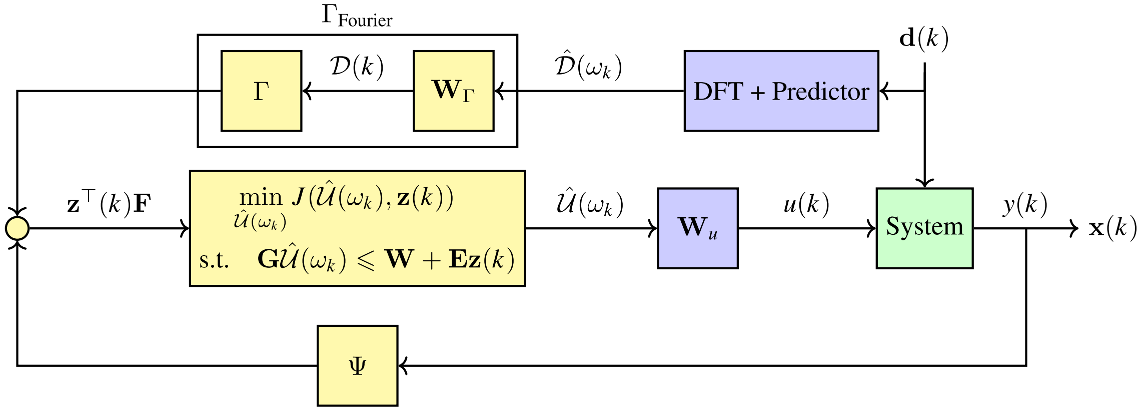
Γ_{Fourier}
Γ
$\mathcal{D}(k)$
$\mathbf{W}_\Gamma$
$\hat{\mathcal{D}}(\omega_k)$
DFT + Predictor
$\mathbf{d}(k)$
$\mathbf{z}^\top(k)\mathbf{F}$
$\min_{\hat{\mathcal{U}}(\omega_k)} J(\hat{\mathcal{U}}(\omega_k), \mathbf{z}(k))$
s.t. $\mathbf{G}\hat{\mathcal{U}}(\omega_k) \leqslant \mathbf{W} + \mathbf{E}\mathbf{z}(k)$
$\hat{\mathcal{U}}(\omega_k)$
$\mathbf{W}_u$
$u(k)$
System
$y(k)$
$\mathbf{x}(k)$
Ψ

FIGURE 7. LS³MPC closed-loop block diagram.

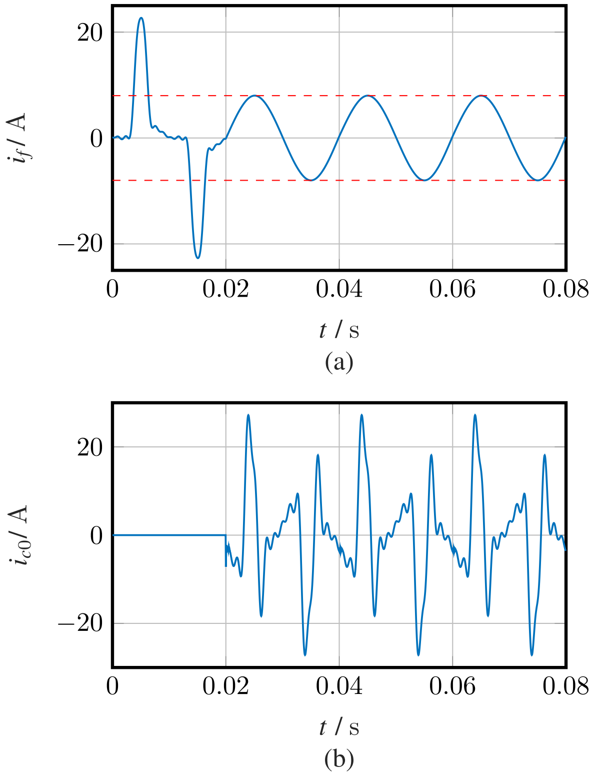

FIGURE 8. LS³MPC closed-loop simulation feeder line current i_f in (a) and control input ideal compensation current i_{c0} in (b).

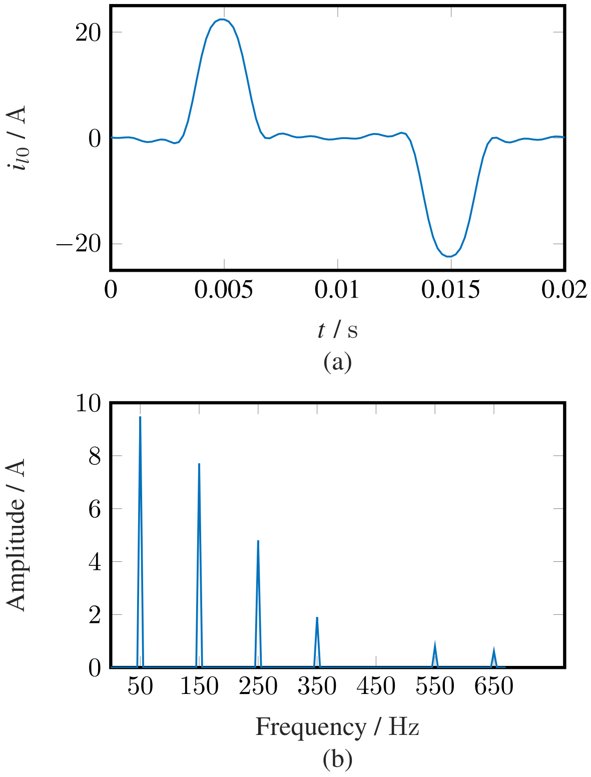

FIGURE 9. Nonlinear load i_{l0} rectifier disturbance signal in time (a) and frequency (b) domain.

the third harmonic of i_{l0}, leads to $\mathsf{T} \in \mathbb{R}^{41\times41\times5\times5\times4\times11}$, i.e., 1.85×10^6 elements, or 4.2×10^4 function matrices $\mathbf{K}_j$ corresponding to each orthotope region to be computed. The function matrix computations were performed in a consumer grade laptop with an Intel® Core™ i5-6300U CPU running at 2.5 GHz and 8 GB of RAM. Each $\mathbf{K}_j$ took on average 80.4×10^{-3} s, which is roughly 56 minutes for computing all the function matrices in this setup.

Once all regions of T are computed, the explicit LS³MPC closed-loop can be established, as seen in the block diagram in Fig. 10. Here, the appropriate function matrix $\mathbf{K}_j$ is selected based on the current $\mathbf{x}(k)$ and $\tilde{\mathcal{D}}(\omega_k)$ values. Once again, direct access to the states $\mathbf{x}(k)$ is assumed, and the same DFT and predictor concepts from the LS³MPC closed-loop case are kept. Fig. 11 shows the results considering that the target feeder line current i_f (solid blue line) was constrained to ±8 A (red dashed line), and the input weighting matrix $\mathbf{R}$ was set to be diagonal with all entries set to 1×10^{-8}. In this case, the compensating action of the explicit LS³MPC reached a THD of under 4.13 %. Naturally, a finer mesh would lead to even better results but at the cost of storage, as already discussed.

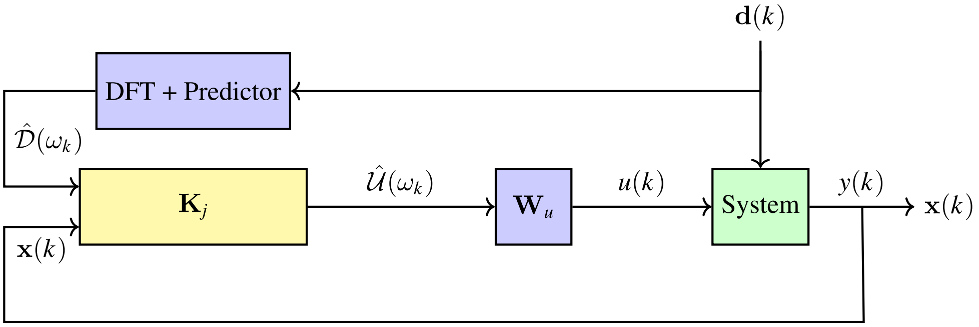

FIGURE 10. Explicit LS^{3}MPC closed-loop block diagram.

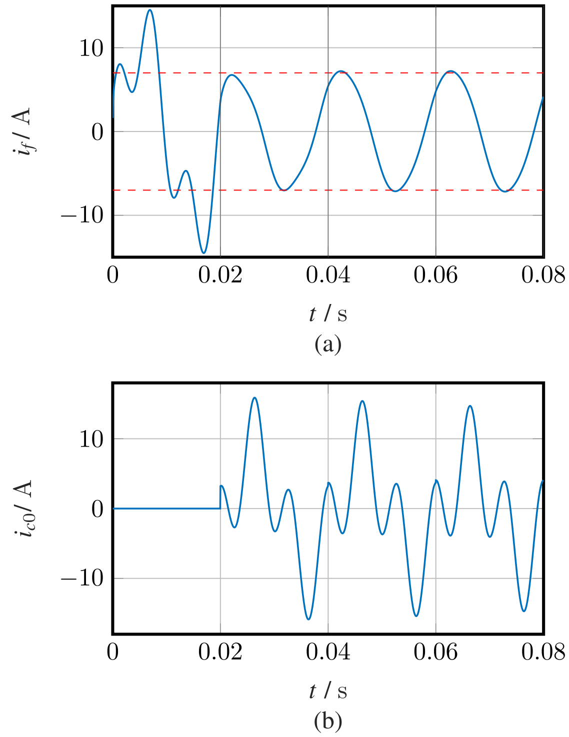

FIGURE 11. Explicit LS^{3}MPC closed-loop simulation feeder line current i_f in (a) and control input ideal compensation current i_{c0} in (b).

D. REDUCTION OF STORAGE DEMAND

Despite the simplifications introduced by the end of subsubsection IV-C to reduce the number of dimensions of T and their size, 1.85×10^6 elements is still quite numerous. As explained in subsubsection III-C, since T is a tensor, it is possible to approximate it with a low-rank CP decomposition. For this example, the tensor T was decomposed using the tensorlab toolbox [34]. Different rank approximations were tested; Fig. 12 shows the compensation results for approximations with rank $R = 35$, $R = 240$, and $R = 500$, which portray the trade-off between accuracy and storage demand as the rank increases. Once again, the target feeder line current i_f was constrained to ±8 A (red dashed line), and the input weighting matrix $\mathbf{R}$ was set to be diagonal with all entries set to 1×10^{-8}.

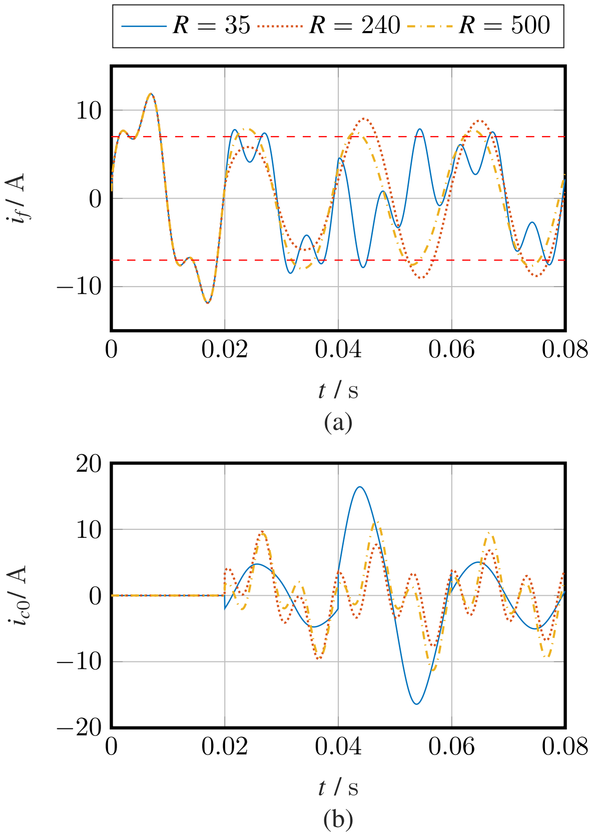

FIGURE 12. Explicit LS^{3}MPC closed-loop simulation feeder line current i_f in (a) and control input ideal compensation current i_{c0} in (b) with stored tensors with different rank approximations.

From Fig. 12, it is apparent that a rank of 35 is not sufficient to control the system. A rank of 240, on the other hand, is already shaping the signal quite well, but the constraint of not exceeding the upper and lower bounds is violated. The controller using the rank 500 approximation can keep the results within the constraints while providing a good compensating action.

Regarding the number of elements to be stored, the rank 500 approximation needs to store only 5.35×10^4 elements, considerably less than the full tensor at 1.85×10^6 elements, a reduction of more than 97 %.

TABLE 2. THD of the current i_f for the simulation of the explicit LS^{3}MPC with decomposed tensors with different ranks.

	THD$_{\%}$		
Tensor rank R	period 2	period 3	period 4
35	43.7502 %	96.3529 %	54.5837 %
240	5.7313 %	1.8838 %	3.5363 %
500	5.5127 %	2.5367 %	5.1439 %

In Table 2, the THD of the current i_f is given for the first periods using decomposed tensors with different ranks in the explicit LS^{3}MPC. Regarding the THD results performance, the efficacy of the controller using the decomposed tensor with rank 240 and rank 500 is clear.

It is also important to notice that all these low-rank approximations cannot formally ensure any of the closed-loop properties of the original MPC problem, e.g., stability or robustness. This connection was already formally severed by the orthotope approximation and is further diminished due to the decomposition. It is easy to notice from the results that the higher the rank, the higher the closed-loop fidelity (similarly for the mesh size fineness). However, a formal derivation of bounds on the induced errors and deviation from the original closed-loop behavior is yet to be developed and is part of future research as the method continues to mature.

V. CONCLUSION

The explicit LS^{3}MPC based on tensor decompositions combines the linear state signal shaping MPC approach with a multidimensional extension to the offline formulation of general explicit MPC theory. It was shown that with the described meshing approach, it is possible to store the relevant information for each orthotope, i.e., the meshed space of states and disturbances in a tensor. This approach not only reduces the number of variables required to store the relevant regions but grants faster access to the function matrices $\mathbf{K}_j$, as each dimension is uniformly divided, thus trivializing the point location problem. Moreover, switching some key variables to the frequency domain makes the tensor size independent of the prediction horizon, focusing on the number of harmonics relevant for the application. More importantly, the computed tensor can be approximated by a low-rank CP tensor, which leads to more efficient tensor operations and reduces the memory footprint considerably. Without the reduction brought by tensor decomposition, the chosen meshing approach would be unviable due to the curse of dimensionality.

The power system application example shows that for a simple circuit consisting of a grid connection, an SAPF, and a nonlinear load, the explicit LS^{3}MPC can compensate the harmonics introduced by the nonlinear load. Simulation results for reduced rank tensors proved to be satisfactory regarding the THD and the compliance of the constraints. It was shown that for a rank of 500, a reduction of the storage demand by more than 97 % still produces acceptable harmonic compensation. This showcased the expected trade-off against approximation error, constraint violations, and region discontinuities. These results are promising for the practical implementation of the LS^{3}MPC approach.

Future work will focus on further improving the main limiting factor of the approach, i.e, the offline calculation of the full tensor and its decomposition. In this context, many approaches are possible, like exploring using incomplete tensor formulations or combining the CP approximation optimization problem with the MPC problem to find a low-rank CP decomposed tensor directly instead. Later research will focus on the derivation of bounds for the closed-loop behavior that can lead to formal certificates on the stability and robustness of the approach. Additionally, numerical simulation comparisons are to be carried out with competitive methods in literature.

APPENDIX
TOTAL HARMONIC DISTORTION

Definition 3 (Total harmonic distortion): The total harmonic distortion (THD) of a signal I is given as

$$\mathrm{THD} = \frac{\sqrt{\sum_{h=2}^{\infty} \left(I^{(h)}\right)^2}}{I^{(1)}} \cdot 100\,\% \,, \tag{36}$$

where $I^{(h)}$ denotes the root mean square (RMS) or the amplitude of the h^{th} harmonic of the signal I, [36].

REFERENCES

[1] B. Singh, *Power Quality: Problems and Mitigation Techniques*, 1st ed. Hoboken, NJ, USA: Wiley, 2015.

[2] M. Mehrasa, E. Pouresmaeil, S. Zabihi, E. M. G. Rodrigues, and J. P. S. Catalão, "A control strategy for the stable operation of shunt active power filters in power grids," *Energy*, vol. 96, pp. 325–334, Feb. 2016.

[3] J. Fei and T. Wang, "Adaptive fuzzy-neural-network based on RBFNN control for active power filter," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 5, pp. 1139–1150, May 2019.

[4] K. Weihe, C. C. Yáñez, G. Pangalos, and G. Lichtenberg, "Constrained linear state signal shaping model predictive control for harmonic compensation in power systems," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6937–6942, 2020.

[5] K. Weihe, C. C. Yáñez, G. Pangalos, and G. Lichtenberg, "Comparison of linear state signal shaping model predictive control with classical concepts for active power filter design," in *Simultech*. Setúbal, Portugal: Science and Technology Publications, 2018, pp. 167–174.

[6] V. V. Naik and A. Bemporad, "Exact and heuristic methods with warm-start for embedded mixed-integer quadratic programming based on accelerated dual gradient projection," 2021, *arXiv:2101.09264*.

[7] G. Cimini, D. Bernardini, S. Levijoki, and A. Bemporad, "Embedded model predictive control with certified real-time optimization for synchronous motors," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 2, pp. 893–900, Mar. 2021.

[8] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, Jan. 2002.

[9] L. Galč íková and J. Oravec, "Fixed complexity solution of partial explicit MPC," *Comput. Chem. Eng.*, vol. 157, Jan. 2022, Art. no. 107606.

[10] T. Geyer, F. D. Torrisi, and M. Morari, "Optimal complexity reduction of polyhedral piecewise affine systems," *Automatica*, vol. 44, no. 7, pp. 1728–1740, Jul. 2008.

[11] C. Wen, X. Ma, and B. E. Ydstie, "Analytical expression of explicit MPC solution via lattice piecewise-affine function," *Automatica*, vol. 45, no. 4, pp. 910–917, Apr. 2009.

[12] M. Kvasnica, B. Takács, J. Holaza, and S. Di Cairano, "On region-free explicit model predictive control," in *Proc. 54th IEEE Conf. Decis. Control (CDC)*, Dec. 2015, pp. 3669–3674.

[13] A. Gupta, S. Bhartiya, and P. S. V. Nataraj, "A novel approach to multiparametric quadratic programming," *Automatica*, vol. 47, no. 9, pp. 2112–2117, Sep. 2011.

[14] R. Mitze, M. Kvasnica, and M. Mönnigmann, "Exploiting symmetries in active set enumeration for constrained linear–quadratic optimal control," *Automatica*, vol. 151, May 2023, Art. no. 110900.

[15] F. Bayat, T. A. Johansen, and A. A. Jalali, "Using hash tables to manage the time-storage complexity in a point location problem: Application to explicit model predictive control," *Automatica*, vol. 47, no. 3, pp. 571–577, Mar. 2011.

[16] F. Bayat, T. A. Johansen, and A. A. Jalali, "Flexible piecewise function evaluation methods based on truncated binary search trees and lattice representation in explicit MPC," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 3, pp. 632–640, May 2012.

[17] J. Holaza, J. Oravec, M. Kvasnica, R. Dyrska, M. Mönnigmann, and M. Fikar, "Accelerating explicit model predictive control by constraint sorting," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 11356–11361, 2020.

[18] A. Bemporad and C. Filippi, "Suboptimal explicit receding horizon control via approximate multiparametric quadratic programming," *J. Optim. Theory Appl.*, vol. 117, no. 1, pp. 9–38, Apr. 2003.

[19] T. A. Johansen and A. Grancharova, "Approximate explicit constrained linear model predictive control via orthogonal search tree," *IEEE Trans. Autom. Control*, vol. 48, no. 5, pp. 810–815, May 2003.

[20] F. Scibilia, S. Olaru, and M. Hovd, "Approximate explicit linear MPC via Delaunay tessellation," in *Proc. Eur. Control Conf. (ECC)*, Aug. 2009, pp. 2833–2838.

[21] A. Gersnoviez, M. Brox, and I. Baturone, "High-speed and low-cost implementation of explicit model predictive controllers," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 2, pp. 647–662, Mar. 2019.

[22] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017.

[23] G. Lichtenberg, G. Pangalos, C. Cateriano Yáñez, A. Luxa, N. Jöres, L. Schnelle, and C. Kaufmann, "Implicit multilinear modeling: An introduction with application to energy systems," *At Automatisierungstechnik*, vol. 70, no. 1, pp. 13–30, Jan. 2022.

[24] S. Summers, C. N. Jones, J. Lygeros, and M. Morari, "A multiresolution approximation method for fast explicit model predictive control," *IEEE Trans. Autom. Control*, vol. 56, no. 11, pp. 2530–2541, Nov. 2011.

[25] J. M. Maciejowski, *Predictive Control: With Constraints*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[26] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "OSQP: An operator splitting solver for quadratic programs," *Math. Program. Comput.*, vol. 12, no. 4, pp. 637–672, Dec. 2020.

[27] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[28] C. C. Yáñez, G. Pangalos, and G. Lichtenberg, "An approach to linear state signal shaping by quadratic model predictive control," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2018, pp. 1–6.

[29] J.-H. Meyer, "Expliziter modellpradiktiver regler zur linearen signalformung," M.S. thesis, Fac. Eng. Comput. Sci., Dept. Inf. Elect. Eng., Hamburg Univ. Appl. Sci., Hamburg, Germany, 2021.

[30] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, Aug. 2009.

[31] I. V. Oseledets, "Tensor-train decomposition," *SIAM J. Scientific Comput.*, vol. 33, no. 5, pp. 2295–2317, Jan. 2011.

[32] L. Grasedyck, "Hierarchical singular value decomposition of tensors," *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 4, pp. 2029–2054, Jan. 2010.

[33] L. Grasedyck, D. Kressner, and C. Tobler, "A literature survey of low-rank tensor approximation techniques," *GAMM-Mitteilungen*, vol. 36, no. 1, pp. 53–78, Aug. 2013.

[34] N. Vervliet, O. Debals, L. Sorber, M. van Barel, and L. de Lathauwer. (2016). *Tensorlab 3.0*. [Online]. Available: https://www.tensorlab.net

[35] B. Fornberg, "Generation of finite difference formulas on arbitrarily spaced grids," *Math. Comput.*, vol. 51, no. 184, p. 699, Oct. 1988.

[36] D. Shmilovitz, "On the definition of total harmonic distortion and its effect on measurement interpretation," *IEEE Trans. Power Del.*, vol. 20, no. 1, pp. 526–528, Jan. 2005.

CARLOS CATERIANO YÁÑEZ was born in Arequipa, Peru, in 1990. He received the B.Eng. degree in industrial and systems engineering from the University of Piura, Lima, Peru, in 2012, and the M.Eng. degree in renewable energy systems from Hamburg University of Applied Sciences, Hamburg, Germany, in 2017. He is currently pursuing the Ph.D. degree in automation, robotics, and industrial computer science with Universitat Politècnica de València, Valencia, Spain, in cooperation with Hamburg University of Applied Sciences, Hamburg, Germany.

Since 2017, he has been with Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V., Hamburg. His research interests include model predictive control and its applications in the field of power systems integrated with renewable energy sources.

GEORG PANGALOS received the Dipl.-Ing. degree in electrical engineering and the Ph.D. degree in control engineering from Hamburg University of Technology, in 2011 and 2015 respectively.

Since 2015, he has been with Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V., Hamburg, Germany. His research interests include renewable energy integration and energy systems transition.

JAN-HENRIK MEYER received the bachelor's degree in electrical engineering and information technology and the master's degree in automation technology from Hamburg University of Applied Sciences, in 2018 and 2021, respectively. For his master's thesis, he was employed with the Fraunhofer Institute ISIT, Hamburg, where he conducted research in the field of energy and control technology. He is currently with the Automotive Sector, Panasonic Industrial Devices Europe, Lüneburg.

GERWALD LICHTENBERG received the Diploma degree in physics from the University of Hamburg, in 1992, and the Ph.D. and Habilitation degrees in control engineering from Hamburg University of Technology, in 1998 and 2012, respectively.

Currently, he is a Professor of physics and control systems with the Faculty Life Sciences, Hamburg University of Applied Sciences. His research interests include model-based and learning control and fault diagnosis of complex systems, such as local energy networks, building systems, and particle accelerators. The methodological focus of his work is on tensor decomposition methods and multilinear models.

JAVIER SANCHIS SÁEZ received the B.Sc. and Ph.D. degrees in computer science from Universitat Politècnica de València, Spain, in 1993 and 2002, respectively. He is a Professor with the Department of Systems Engineering and Control, Universitat Politècnica de València. He has more than 15 years of experience in applying soft computing methods to engineering problems related with process control and optimization. His main research interests include multivariable predictive control, constrained process control, and computational intelligence methods for systems engineering and process control.