Hierarchical Distributed ADMM for Predictive Control with Applications in Power Networks

Philipp Braun[1,2], Timm Faulwasser[3], Lars Grüne[1], Christopher M. Kellett[2],
Steven R. Weller[2] and Karl Worthmann[4]

Abstract—We present a variant of the alternating direction method of multipliers (ADMM) that can be implemented in a hierarchical distributed fashion for large-scale systems where the coupling between subsystems occurs in a structured way in the cost function. We show that this ADMM algorithm can be embedded in a model predictive control (MPC) implementation and subsequently apply this to three battery scheduling problems in the context of residential microgrid electricity networks.

Index Terms—Large-scale systems, Alternating direction method of multipliers (ADMM), Model predictive control (MPC), Optimal scheduling, Smart grid.

I. INTRODUCTION

The optimal control of large-scale cyber-physical systems presents several difficulties including curse of dimensionality issues in solving large optimization problems, but also accounting for security concerns (e.g., sharing sensitive information between subsystems) or facilitating plug-and-play operation whereby new subsystems can be added without requiring a coordinating agent to have detailed information about the new subsystems.

One technique for coping with the curse of dimensionality is to resort to distributed optimization algorithms when the structure of the problem and systems allow devolving the optimization from a central point to the individual subsystems [1], [2]. With the interest in large-scale multi-agent systems, many distributed optimization algorithms have been recently proposed; e.g., [3], [4], [5] and the references therein.

Avoiding the need to share detailed information amongst subsystems is more difficult in a completely distributed fashion. However, as proposed in [5], the introduction of a hierarchical structure with a trusted central entity that can aggregate information and then broadcast only the aggregated information alleviates the need for subsystems to share detailed information with their neighbors.

In this paper, we present an alternative hierarchical distributed optimization algorithm to that proposed in [5]. Here, we propose a variant of the well-known alternating direction method of multipliers (ADMM) algorithm [6]. For the class of linear systems and a structured cost function, this ADMM variant can be decomposed so that a central entity needs only solve an optimization problem that is independent of the number of subsystems and does not depend on detailed information about the subsystems, while each subsystem solves an optimization problem that depends only on local information and a single quantity broadcast from the central entity. Consequently, this ADMM variant addresses the above concerns that frequently arise in coordinating large-scale systems.

As an application of the above-described ADMM variant, we consider three problems related to the optimal scheduling of energy storage in a residential-scale electricity network. Similar to the work in [7], [5], we consider the problem of smoothing the energy demand from a group of residences equipped with energy storage. However, the proposed ADMM variant is somewhat more flexible than the approach considered in [7], [5] and we also address two additional problems.

The first looks at time-varying tube constraints, that allow the grid operator to steer demand within reasonable limits. Such an approach may be useful for the energy provider in choosing the quantity and type of external generation resources. The second problem looks at computing and/or implementing a maximal islanding time for a microgrid connected to a wider grid via a point of common coupling (see [8] and the references therein). In this context, a grid operator can forecast how long a microgrid can be disconnected from the larger grid, while still maintaining sufficient local supply to meet demand. Implemented in a predictive control context, the microgrid can collectively store energy prior to the disconnection time, so as to maximize the time it can operate independently.

The paper is structured as follows: In Section II we formulate an optimal control problem in the form of a nonlinear optimization problem (NLP) for linear time-varying discrete-time control systems coupled through a set of variables in the objective function. In Section III we introduce a hierarchical distributed optimization algorithm and embed it in a receding horizon framework. In Section IV we recall the system dynamics of an electricity network satisfying the assumptions of the dynamics introduced in Section II. The flexibility of the hierarchical distributed optimization algorithm with respect to the objective in the context of model predictive control is demonstrated in Section V based on the example of the microgrid. The paper concludes in Section VI.

II. OPTIMAL CONTROL FORMULATION FOR A NETWORK OF LINEAR SYSTEMS

We consider a network of $\mathcal{I} \in \mathbb{N}$ linear time-varying discrete-time systems

$$\Sigma_i : \begin{cases} x_i(k+1) = A_i(k)x_i(k) + B_i(k)u_i(k) + v_i(k) \\ \quad z_i(k) = E_i(k)x_i(k) + F_i(k)u_i(k) + w_i(k) \end{cases} \quad (1)$$

where $x_i \in \mathbb{X}_i \subset \mathbb{R}^{n_i}$, $u_i \in \mathbb{U}_i \subset \mathbb{R}^{m_i}$, $z_i \in \mathbb{R}^p$ are the state, input, and coupling variables, respectively, $(v_i(k))_{k \in \mathbb{N}} \subset \mathbb{R}^{n_i}$ and $(w_i(k))_{k \in \mathbb{N}} \subset \mathbb{R}^p$ are known exogenous signals, and $A_i(k) \in \mathbb{R}^{n_i \times n_i}$, $B_i(k) \in \mathbb{R}^{n_i \times m_i}$, $E_i(k) \in \mathbb{R}^{p \times n_i}$, $F_i(k) \in \mathbb{R}^{p \times m_i}$ are time-dependent matrices defining the system dynamics for all $k \in \mathbb{N}$ and for all $i \in \mathbb{N}_\mathcal{I} := \{1, \dots, \mathcal{I}\}$.

As shown in Figure 1, systems Σ_i are coupled through the variables z_i, $i \in \mathbb{N}_\mathcal{I}$, leading to the overall system dynamics

$$\overline{\Sigma} : \begin{cases} x(k+1) = A(k)x(k) + B(k)u(k) + v(k) \\ \overline{z}(k) = \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} (E_i(k)x_i(k) + F_i(k)u_i(k) + w_i(k)) \end{cases}$$

with $x = (x_1^T, \dots, x_\mathcal{I}^T)^T$, $u = (u_1^T, \dots, u_\mathcal{I}^T)^T$, $v = (v_1^T, \dots, v_\mathcal{I}^T)^T$, $w = (w_1^T, \dots, w_\mathcal{I}^T)^T$, and the averaged coupling variable $\overline{z} = \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} z_i$. The definition of matrices $A(k)$, $B(k)$ follow immediately from the definition of the individual dynamics Σ_i. Additionally we use the notation $\mathbb{X} = \mathbb{X}_1 \times \cdots \times \mathbb{X}_\mathcal{I}$ and $\mathbb{U} = \mathbb{U}_1 \times \cdots \times \mathbb{U}_\mathcal{I}$ to rewrite the state and input constraints. Throughout this paper we assume
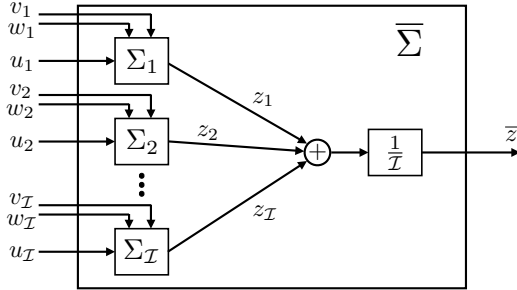

Fig. 1. Visualization of the individual systems Σ_i and the overall system $\overline{\Sigma}$.

that the states x_i are known or at least observable.

For a given prediction horizon $N \in \mathbb{N}$ and a fixed time index $k \in \mathbb{N}$ we wish to solve the following (receding horizon) optimal control problem:

$$\min_{(u_i, x_i, z_i)} \sum_{i=1}^{\mathcal{I}} \Big(\sum_{j=k}^{k+N-1} \ell_i(z_i(j)) \Big) + \sum_{j=k}^{k+N-1} \overline{\ell}\,(\overline{z}(j)) \quad (2a)$$

$$\text{s.t. } x_i(k) = x_i^0 \quad (2b)$$
$$x_i(j+1) = A_i(j)x_i(j) + B_i(j)u_i(j) + v_i(j) \quad (2c)$$
$$z_i(j) = E_i(j)x_i(j) + F_i(j)u_i(j) + w_i(j) \quad (2d)$$
$$x_i(j+1) \in \mathbb{X}_i, \quad u_i(j) \in \mathbb{U}_i, \quad (2e)$$
$$\forall\, j = k, \dots, k+N-1, \ i = 1, \dots, \mathcal{I}.$$

Here, $\ell_i : \mathbb{R}^p \to \mathbb{R}$ $(i \in \mathbb{N}_\mathcal{I})$ and $\overline{\ell} : \mathbb{R}^p \to \mathbb{R}$ define running costs which penalize the deviation from a desired reference for the individual and averaged coupling variables, respectively.

The optimal control problem (2) defines a finite dimensional NLP in the unknowns x_i, u_i, and z_i, which can be solved

by standard optimization software if the number of systems $\mathcal{I}$ is reasonably small. Otherwise, a distributed computation is needed to ensure scalability and flexibility with respect to the number of subsystems $\mathcal{I}$. The main difficulty in (2) is the coupling in the objective function through the variables $\overline{z}$. One solution to this difficulty is to use a distributed optimization approach that solves the NLPs only at the subsystem level, Σ_i, rather than at the overall system level, $\overline{\Sigma}$. To this end, in Section III, we summarize the alternating direction method of multipliers (ADMM) and describe a distributed computation implementation.

To present the ADMM algorithm in Section III, two preparatory steps are required. Firstly, we rewrite the constraints (2b)–(2e) in a more compact form. Secondly, we embed the optimization problem (2) into a more general one, which facilitates a desirable decomposition.

For a given time index $k \in \mathbb{N}$ and a prediction horizon $N \in \mathbb{N}$ we use the notation

$$\mathbf{z}(k) := (\mathbf{z}_1(k)^T, \mathbf{z}_2(k)^T, \dots, \mathbf{z}_\mathcal{I}(k)^T)^T,$$

where $\mathbf{z}_i(k) := (z_i(k)^T, \dots, z_i(k+N-1)^T)^T$ is defined componentwise. To simplify the notation we use $\mathbf{z} := \mathbf{z}(k)$ and $\mathbf{z}_i := \mathbf{z}_i(k)$ whenever the k dependency is clear from the context. We use $\bar{\cdot}$ to denote the average of a vector; e.g.,

$$\overline{\mathbf{z}}(k) := \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \mathbf{z}_i(k).$$

The same notation is used for the other variables such as x and u. At a fixed time index $k \in \mathbb{N}$, for given initial values $x_i(k) \in \mathbb{X}_i$, we define the set of feasible trajectories of system i, $i \in \mathbb{N}_\mathcal{I}$, as

$$\mathbb{D}_i = \left\{ \mathbf{z}_i \in \mathbb{R}^{pN} \left| \begin{array}{l} \exists\, x_i(j+1) \in \mathbb{X}_i, u_i(j) \in \mathbb{U}_i \text{ satisfying} \\ (2b)\text{--}(2d) \text{ for } j = k, \dots, k+N-1 \end{array} \right. \right\}. \quad (3)$$

Even though an explicit expression of the sets $\mathbb{D}_i$, $i = 1, \dots, \mathcal{I}$ cannot be given, it is easy to see that they satisfy the following convexity property if the sets $\mathbb{X}_i$ and $\mathbb{U}_i$ are convex.

Proposition II.1. *Let the sets $\mathbb{U}_i$ and $\mathbb{X}_i$ be convex, $\mathbb{X}_i$ be closed, and $\mathbb{U}_i$ be compact. Then, for given $k, N \in \mathbb{N}$, $x_i^0 \in \mathbb{X}_i$, $(v_i(j))_{j \in \mathbb{N}} \subset \mathbb{R}^{n_i}$, and $(w_i(j))_{j \in \mathbb{N}} \subset \mathbb{R}^{m_i}$, the set $\mathbb{D}_i$ defined in (3) is convex and compact.*

By introducing the auxiliary variable $\mathbf{s} \in \mathbb{R}^M$, $M \in \mathbb{N}$, and the set $\mathbb{S} \subset \mathbb{R}^{pN} \times \mathbb{R}^M$, the optimization problem (2) can be generalized to

$$(\mathbf{z}^\star, \mathbf{s}^\star) \in \underset{(\mathbf{z}, \mathbf{s}) \in \mathbb{S}}{\operatorname{argmin}} \sum_{i=1}^{\mathcal{I}} f_i(\mathbf{z}_i) + g(\overline{\mathbf{z}}) + h(\mathbf{s}) \quad (4)$$
$$\text{s.t. } \mathbf{z}_i \in \mathbb{D}_i \text{ for all } i = 1, \dots, \mathcal{I}.$$

The functions $f_i : \mathbb{R}^{pN} \to \mathbb{R}$, $i = 1, \dots, \mathcal{I}$, and $g : \mathbb{R}^{pN} \to \mathbb{R}$ capture sums over ℓ_i and ℓ. Moreover, we define the overall objective function $K : \mathbb{R}^{\mathcal{I}pN} \times \mathbb{R}^{pN} \times \mathbb{R}^M \to \mathbb{R}$,

$$K(\mathbf{z}, \overline{\mathbf{z}}, \mathbf{s}) = \sum_{i=1}^{\mathcal{I}} f_i(\mathbf{z}_i) + g(\overline{\mathbf{z}}) + h(\mathbf{s}). \quad (5)$$

The function $h : \mathbb{R}^M \to \mathbb{R}$ and the set $\mathbb{S}$ offer an additional degree of freedom in the objective function and in the

constraints. Observe that in the optimization problem (2) the variables u_i, x_i, z_i are optimized whereas in the optimization problem (4) the variable $\mathbf{z}$ is optimized. Using the variables $\mathbf{z}$ will simplify the notation of the optimization problems in the upcoming sections. Due to the linear dynamics (1), given an optimal solution $\mathbf{z}^\star$, an optimal solution $\mathbf{x}^\star$ and $\mathbf{u}^\star$ can be easily constructed. However, the optimal solution $\mathbf{z}^\star$, as well as corresponding $\mathbf{x}^\star$ and $\mathbf{u}^\star$, are in general not unique.

III. RECEDING HORIZON CONTROL USING THE ALTERNATING DIRECTION METHOD OF MULTIPLIERS

In this section we describe ADMM for the solution of optimization problems of the form (4) which can be found, for example, in detail in [6] and [2]. After the basic ADMM scheme is described, we show how the formulation can be simplified such that the number of unknowns in the optimization problem can be reduced significantly. In the last part of this section we show how the ADMM algorithm can be embedded efficiently in a model predictive control (MPC) implementation.

A. The alternating direction method of multipliers

The ADMM scheme solves optimization problems in a distributed way and is based on the ideas of dual decomposition and dual ascent. We rewrite the optimization problem (4) by introducing the auxiliary variables $\mathbf{a}_i$, $i = 1, \ldots, \mathcal{I}$, and $\mathbf{a} = (\mathbf{a}_1^T, \mathbf{a}_2^T, \ldots, \mathbf{a}_\mathcal{I}^T)^T$ and $\overline{\mathbf{a}}$ are defined analogously to $\mathbf{z}$ and $\overline{\mathbf{z}}$; i.e., optimization problem (4) can be written as follows:

$$\left.\begin{aligned} \min_{(\mathbf{z},\mathbf{a},\mathbf{s})} \quad & K(\mathbf{z},\overline{\mathbf{a}},\mathbf{s}) \\ \text{s.t.} \quad & (\overline{\mathbf{a}},\mathbf{s}) \in \mathbb{S} \\ & \mathbf{z}_i \in \mathbb{D}_i, \quad \mathbf{z}_i - \mathbf{a}_i = 0, \quad \forall\, i = 1,\ldots,\mathcal{I} \end{aligned}\right\} . \quad (6)$$

The augmented Lagrangian $\mathcal{L}_\rho : \mathbb{R}^{\mathcal{I}pN} \times \mathbb{R}^{\mathcal{I}pN} \times \mathbb{R}^M \times \mathbb{R}^{\mathcal{I}pN} \to \mathbb{R}$ with respect to (6) is given by

$$\mathcal{L}_\rho(\mathbf{z},\mathbf{a},\mathbf{s},\lambda) = K(\mathbf{z},\overline{\mathbf{a}},\mathbf{s}) + \lambda^T(\mathbf{z} - \mathbf{a}) + \frac{\rho}{2}\|\mathbf{z}-\mathbf{a}\|^2 \quad (7)$$

with Lagrange multipliers $\lambda = (\lambda_1^T, \lambda_2^T, \ldots, \lambda_\mathcal{I}^T)^T$ and penalty term $\rho \in \mathbb{R}_{>0}$. For $\rho = 0$, we obtain the usual definition of the Lagrangian.

ADMM is an iterative scheme to solve problem (4) (equivalently (6)). Both the update step of each iteration and convergence properties are presented in the following theorem, see [6, Appendix A] for a proof and an in-depth detailed analysis of the convergence properties of the ADMM scheme (8).

Theorem III.1. *Let the extended-real-valued functions f_i, $i \in \mathbb{N}_\mathcal{I}$, g, and h be closed, convex, and proper[1]. Suppose, in addition, that a saddle point $(\mathbf{z}^\star, \mathbf{a}^\star, \mathbf{s}^\star, \lambda^\star)$ of the (unaugmented) Lagrangian $\mathcal{L}_0$ exists, meaning that*

$$\mathcal{L}_0(\mathbf{z}^\star, \mathbf{a}^\star, \mathbf{s}^\star, \lambda) \leq \mathcal{L}_0(\mathbf{z}^\star, \mathbf{a}^\star, \mathbf{s}^\star, \lambda^\star) \leq \mathcal{L}_0(\mathbf{z}, \mathbf{a}, \mathbf{s}, \lambda^\star)$$

[1] An extended-real-valued function $f : D \to \mathbb{R} \cup \{\pm\infty\}$ is called proper if its effective domain $\{x \in D | f(x) \in \mathbb{R}\}$ is nonempty and there does not exist $x \in D$ such that $f(x) = -\infty$.

holds for all λ and all $(\mathbf{z},\mathbf{a},\mathbf{s})$. Then, the iteration scheme

$$\mathbf{z}_i^{\ell+1} := \operatorname*{argmin}_{\mathbf{z}_i \in \mathbb{D}_i} \mathcal{L}_\rho\left(\mathbf{z}, \mathbf{a}^\ell, \mathbf{s}^\ell, \lambda^\ell\right) \quad (8a)$$

$$(\mathbf{a}^{\ell+1}, \mathbf{s}^{\ell+1}) := \operatorname*{argmin}_{(\overline{\mathbf{a}},\mathbf{s}) \in \mathbb{S}} \mathcal{L}_\rho\left(\mathbf{z}^{\ell+1}, \mathbf{a}, \mathbf{s}, \lambda^\ell\right) \quad (8b)$$

$$\lambda^{\ell+1} := \lambda^\ell + \rho\left(\mathbf{z}^{\ell+1} - \mathbf{a}^{\ell+1}\right) \quad (8c)$$

satisfies the following convergence properties for any initializations $\mathbf{z}^0, \mathbf{a}^0, \mathbf{s}^0$, and λ^0, and for all fixed stepsizes $\rho > 0$:

(i) *The sequence $(\mathbf{z}^\ell - \mathbf{a}^\ell)_{\ell\in\mathbb{N}}$ converges to zero, ensuring feasibility of the optimization problem (6).*

(ii) *The sequence $(K(\mathbf{z}^\ell, \mathbf{a}^\ell, \mathbf{s}^\ell))_{\ell\in\mathbb{N}}$ converges to the optimal value $K^\star$ of problem (6) for $\ell \to \infty$.*

(iii) *The dual variables λ^ℓ converge to the optimal dual point $\lambda^\star$ for $\ell \to \infty$.*

One class of problems satisfying the assumption of Theorem III.1 are convex functions subject to nonempty, convex, and compact constraints. Thus, in our context, the assumptions of Theorem III.1 are satisfied if the set $\mathbb{S}$ and the sets $\mathbb{D}_i$, $i = 1, \ldots, \mathcal{I}$, are convex and compact (see Proposition II.1) and there exist $\mathbf{z}_i \in \mathbb{D}_i$ for $i = 1, \ldots, \mathcal{I}$, such that $(\overline{\mathbf{z}}, \mathbf{s}) \in \mathbb{S}$. Since convex functions attain their minimum on compact sets, there exists a (possibly non unique) optimal primal solution $(\mathbf{z}^\star, \mathbf{a}^\star, \mathbf{s}^\star)$ if the minimization problem (4) (or equivalently (6)) is feasible. This implies the existence of an optimal dual solution $\lambda^\star$ satisfying the saddle point condition; see, e.g., [2, Appendix C].

B. Distributed Computation of the ADMM Update

The first update (8a) of the ADMM scheme (8) can be computed in a distributed manner due to the separability (in $\mathbf{z}$) of the cost function K given by (5) and due to the definition of the augmented Lagrangian $\mathcal{L}_\rho$ in (7); i.e.,

$$\mathbf{z}_i^{\ell+1} = \operatorname*{argmin}_{\mathbf{z}_i \in \mathbb{D}_i} f_i(\mathbf{z}_i) + \left(\mathbf{z}_i - \mathbf{a}_i^\ell\right)^T \lambda_i^\ell + \frac{\rho}{2}\left\|\mathbf{z}_i - \mathbf{a}_i^\ell\right\|^2 \quad (9)$$

$$= \operatorname*{argmin}_{\mathbf{z}_i \in \mathbb{D}_i} f_i(\mathbf{z}_i) + \frac{\rho}{2}\left\|\rho^{-1}\lambda_i^\ell + \mathbf{z}_i - \mathbf{a}_i^\ell\right\|^2, \quad (10)$$

which can be solved for all i simultaneously.

The NLP (8b) cannot be separated due to the coupling in the function g. However, it is possible to make the number of unknowns in this problem independent of the number of subsystems $\mathcal{I}$ by using the average variables $\overline{\mathbf{a}} \in \mathbb{R}^{pN}$ instead of $\mathbf{a} \in \mathbb{R}^{\mathcal{I}pN}$.

Proposition III.2. *The updates (8b) and (8c) are equivalent to computing*

$$(\overline{\mathbf{a}}^{\ell+1}, \mathbf{s}^{\ell+1}) =$$
$$\operatorname*{argmin}_{(\overline{\mathbf{a}},\mathbf{s}) \in \mathbb{S}} g\left(\overline{\mathbf{a}}\right) + h(\mathbf{s}) + \frac{\rho\mathcal{I}}{2}\|\rho^{-1} \cdot \overline{\lambda}^\ell + \overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}}\|^2 \quad (11)$$

and, then, setting

$$\overline{\lambda}^{\ell+1} := \overline{\lambda}^\ell + \rho(\overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}}^{\ell+1})$$

where $\overline{\lambda} = \frac{1}{\mathcal{I}}\sum_{i=1}^{\mathcal{I}} \lambda_i$.

To prove Proposition III.2, we require the following lemma.

Lemma III.3. *For given* $\mathbf{c}, \mathbf{y}_i \in \mathbb{R}^N$ *(*$i \in \mathbb{N}_{\mathcal{I}}$*) the equality* $\varphi_i^\star = \mathbf{y}_i + \mathbf{c} - \overline{\mathbf{y}}$ *(*$i \in \mathbb{N}_{\mathcal{I}}$*) with* $\overline{\mathbf{y}} = (1/\mathcal{I}) \sum_{i=1}^{\mathcal{I}} \mathbf{y}_i$ *holds for*

$$\varphi^\star := \operatorname*{argmin}_{\varphi_i \in \mathbb{R}^N} \sum_{i=1}^{\mathcal{I}} \|\varphi_i - \mathbf{y}_i\| \quad s.t. \quad \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \varphi_i = \mathbf{c}.$$

Proof. For $\mathbf{y}_i = 0$ ($i \in \mathbb{N}_{\mathcal{I}}$) the triangle inequality implies

$$\|\mathcal{I} \cdot \mathbf{c}\| = \min_{\varphi_i : \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \varphi_i = \mathbf{c}} \left\| \sum_{i=1}^{\mathcal{I}} \varphi_i \right\|$$
$$\leq \min_{\varphi_i : \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \varphi_i = \mathbf{c}} \sum_{i=1}^{\mathcal{I}} \|\varphi_i\|$$

where equality is attained for $\varphi_i = \mathbf{c}$, $i \in \mathbb{N}_{\mathcal{I}}$. For the general case we use the coordinate transformation $\tilde{\varphi}_i = \varphi_i - \mathbf{y}_i$. Then the equality constraint reads

$$\frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \tilde{\varphi}_i = \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \varphi_i - \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \mathbf{y}_i = \mathbf{c} - \overline{\mathbf{y}}$$

which shows the assertion. $\qquad\square$

Based on this lemma we next prove Proposition III.2 following the arguments given in [6, Chapter 7.3].

Proof. We rewrite the augmented Lagrangian $\mathcal{L}_\rho$ from (7) as

$$g(\overline{\mathbf{a}}) + h(\mathbf{s}) + \sum_{i=1}^{\mathcal{I}} \left(f_i(\mathbf{z}_i) + \frac{\rho}{2} \left\| \mathbf{z}_i - \mathbf{a}_i + \frac{\lambda_i}{\rho} \right\|^2 - \frac{\lambda_i^T \lambda_i}{2\rho} \right).$$

Hence, the minimum of problem (8b) is attained in

$$\operatorname*{argmin}_{(\overline{\mathbf{a}}, \mathbf{s}) \in \mathbb{S}} g(\overline{\mathbf{a}}) + h(\mathbf{s}) + \frac{\rho}{2} \sum_{i=1}^{\mathcal{I}} \left\| \frac{\lambda_i^\ell}{\rho} + \mathbf{z}_i^{\ell+1} - \mathbf{a}_i \right\|^2. \quad (12)$$

Applying Lemma III.3 to problem (12) and fixing the variables $(\overline{\mathbf{a}}, \mathbf{s})$, we see that the optimal solution satisfies

$$\rho^{-1} \cdot \overline{\lambda}^\ell + \overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}} = \rho^{-1} \cdot \lambda_i^\ell + \mathbf{z}_i^{\ell+1} - \mathbf{a}_i \quad (13)$$

for all $i \in \mathbb{N}_{\mathcal{I}}$. Hence the minimization problem (12) is equivalent to

$$\operatorname*{argmin}_{(\overline{\mathbf{a}}, \mathbf{s}) \in \mathbb{S}} g(\overline{\mathbf{a}}) + h(\mathbf{s}) + \frac{\rho}{2} \sum_{i=1}^{\mathcal{I}} \left\| \frac{\overline{\lambda}^\ell}{\rho} + \overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}} \right\|^2 \quad (14)$$

where the number of optimization variables is independent of the number of subsystems and the update (11) follows. Furthermore, multiplying (13) by ρ and combining the result with the update of the Lagrange multipliers (8c) yields

$$\lambda_i^{\ell+1} = \lambda_i^\ell + \rho \left(\mathbf{z}_i^{\ell+1} - \mathbf{a}_i^{\ell+1} \right) = \overline{\lambda}^\ell + \rho (\overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}}^{\ell+1});$$

i.e., $\lambda_i^{\ell+1} = \lambda_j^{\ell+1}$ for all $i, j \in \mathbb{N}_{\mathcal{I}}$. Hence also the update of the dual variables can be written in terms of average variables $\overline{\lambda}$ independent of the number of subsystems, which completes the proof. $\qquad\square$

The update (9), see also (8a), of the variable $\mathbf{z}_i$ involves the variable $\mathbf{a}_i$, which is different for each $i \in \mathbb{N}_{\mathcal{I}}$. This variable has to be transmitted to every subsystem individually during the iteration process. To alleviate this communication need, define

$$\Pi^\ell := \rho^{-1} \cdot \overline{\lambda}^\ell + \overline{\mathbf{z}}^\ell - \overline{\mathbf{a}}^\ell.$$

Since $\lambda_i^\ell = \overline{\lambda}^\ell$, (13) can be rewritten as

$$-\mathbf{a}_i^\ell = -\mathbf{z}_i^\ell + \overline{\mathbf{z}}^\ell - \overline{\mathbf{a}}^\ell$$

Thus, if the variable Π^ℓ is known by the individual systems Σ_i, then the update $\mathbf{z}_i^{\ell+1}$ in (10) can be computed by

$$\operatorname*{argmin}_{\mathbf{z}_i \in \mathbb{D}_i} f_i(\mathbf{z}_i) + \frac{\rho}{2} \left\| \mathbf{z}_i - \mathbf{z}_i^\ell + \Pi^\ell \right\|^2 \quad (15)$$

independent of variables specific to subsystem Σ_j, $j \neq i$.

The hierarchical (distributed) Algorithm 1 summarizes the ideas of this section and splits the ADMM iterations into tasks which can be carried out by the individual subsystems Σ_i in parallel and tasks which must be done by the overall system $\overline{\Sigma}$ or a central entity overseeing the coordination process.

Algorithm 1 Hierarchical distributed optimization algorithm

Phase 1 (Subsystem Σ_i, $i \in \mathbb{N}_{\mathcal{I}}$): Receive Π^ℓ.
- Solve the minimization problem

$$\mathbf{z}_i^{\ell+1} := \operatorname*{argmin}_{\mathbf{z}_i \in \mathbb{D}_i} f_i(\mathbf{z}_i) + \frac{\rho}{2} \left\| \mathbf{z}_i - \mathbf{z}_i^\ell + \Pi^\ell \right\|^2$$

and send $\mathbf{z}_i^{\ell+1}$ to the central entity.

Phase 2 (Central Entity $\overline{\Sigma}$): Receive $\mathbf{z}_i^{\ell+1}$, $i = 1, 2, \ldots, \mathcal{I}$.
- Compute the average $\overline{\mathbf{z}}^{\ell+1} = \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \mathbf{z}_i^{\ell+1}$.
- Solve the minimization problem

$$(\overline{\mathbf{a}}^{\ell+1}, \mathbf{s}^{\ell+1}) = \operatorname*{argmin}_{(\overline{\mathbf{a}}, \mathbf{s}) \in \mathbb{S}} \left(g(\overline{\mathbf{a}}) + h(\mathbf{s}) \right. $$
$$\left. + \frac{\rho \cdot \mathcal{I}}{2} \left\| \overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}} + \frac{\overline{\lambda}^\ell}{\rho} \right\|^2 \right).$$

- Update the Lagrange multiplier

$$\overline{\lambda}^{\ell+1} = \overline{\lambda}^\ell + \rho \left(\overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}}^{\ell+1} \right).$$

- Compute and broadcast

$$\Pi^{\ell+1} = \overline{\mathbf{z}}^{\ell+1} - \overline{\mathbf{a}}^{\ell+1} + \frac{\overline{\lambda}^{\ell+1}}{\rho}.$$

Increment the iteration counter ℓ and go to Phase 1.

Algorithm 1 provides several beneficial properties for systems organized as in Figure 1. Firstly, only the parameter Π is transmitted from the overall system $\overline{\Sigma}$ to the subsystems Σ_i, $i \in \mathbb{N}_{\mathcal{I}}$. Hence, possibly sensitive data does not need to be shared between the subsystems. Secondly, both the dimension of Π (communication overhead) and the number of unknowns in the optimization problem of the central entity is independent of the number of subsystems. Thus, the communication overhead and the computational burden are independent of $\mathcal{I}$.

C. Applying ADMM within a receding horizon scheme

In Section II we proposed a finite dimensional optimization problem (2), that can be solved using ADMM as explained in the previous two subsections. The optimization problem (2) is a standard setting for model predictive control (MPC), where the time index k advances. MPC consists of the following steps, which are repeated at every time instant k. Firstly, optimization problem (2) is solved based on the most recent

measurements (state x_i^0 and the exogenous signals $\mathbf{v}(k)$ and $\mathbf{w}(k)$ of the system $\overline{\Sigma}$), see Step 1) and 2) of Algorithm 2. Then, the first element of the computed sequence $\mathbf{u}^\star(k)$ is implemented, see Step 3) of Algorithm 2, before the horizon is shifted forward in time (*receding horizon control*). This process iteratively defines a feedback law. Since in applications both modelling and estimation (and measurement) errors occur, in general, feedback leads to better and more robust dynamical behavior. For a detailed description we refer to the recent monographs [9] and [10].

Algorithm 2 Distributed model predictive control

1) **Initialization:**

   Subsystem Σ_i, ($i \in \mathbb{N}_{\mathcal{I}}$):

   - Define the local objective f_i, measure the initial state $x_i(k) = x_i^0$ and predict the exogenous signals

     $$\mathbf{v}_i(k) = (v_i(k)^T, \ldots, v_i(k+N-1)^T)^T$$
     $$\mathbf{w}_i(k) = (w_i(k)^T, \ldots, w_i(k+N-1)^T)^T.$$

   Central Entity $\overline{\Sigma}$:

   - Define the objective functions g and h.

2) **Distributed optimization:** Apply Algorithm 1 to compute optimal controls and state trajectories

   $$\mathbf{u}_i^\star(k) = (u_i^\star(k)^T, \ldots, u_i^\star(k+N-1)^T)^T,$$
   $$\mathbf{x}_i^\star(k) = (x_i^\star(k+1)^T, \ldots, x_i^\star(k+N)^T)^T,$$
   $$\mathbf{z}_i^\star(k) = (z_i^\star(k)^T, \ldots, z_i^\star(k+N-1)^T)^T,$$

   for $i = 1, \ldots, \mathcal{I}$.

3) Apply $u_i^\star(k)$ for $i = 1, \ldots, \mathcal{I}$ and increment the time index k

---

In the remainder of this section we discuss two advantages of using Algorithm 1 within model predictive control as it is proposed in Algorithm 2.

First, since the individual subsystems Σ_i do not require knowledge of the objective functions g and h, the central entity may modify these functions at every time step k without reference to the subsystems Σ_i. Conversely, each subsystem Σ_i, $i \in \mathbb{N}_{\mathcal{I}}$, is free to change its system dynamics (1), its (local) constraints $\mathbb{X}_i$, $\mathbb{U}_i$, and its objective function f_i. This yields significant flexibility and allows an adaptation of the objectives during the runtime of the algorithm and a plug-and-play installation in a real world application.

Second, if the objective functions and the system dynamics are not changed between two consecutive time steps k and $k+1$ the NLPs solved at both time steps are very similar. Thus, the average number of iterations necessary to obtain an optimal solution using Algorithm 1 can be reduced significantly with a so-called *warm-start* technique; see for example [10, Ch. 10.5]. In our context a natural way to include warm-start in Algorithm 2 is to initialize $\overline{\lambda}^0$ and Π^0 at time step $k+1$ based on $\overline{\lambda}$ and Π from time k; i.e., we define

$$\overline{\lambda}_{k+1}^0 := (\overline{\lambda}_k^\star(2)^T, \overline{\lambda}_k^\star(3)^T, \ldots, \overline{\lambda}_k^\star(N)^T, 0)^T$$
$$\Pi_{k+1}^0 := (\Pi_k^\star(2)^T, \Pi_k^\star(3)^T, \ldots, \Pi_k^\star(N)^T, 0)^T$$

where

$$\overline{\lambda}_k^\star := (\overline{\lambda}_k^\star(1)^T, \overline{\lambda}_k^\star(2)^T, \ldots, \overline{\lambda}_k^\star(N)^T)^T$$
$$\Pi_k^\star := (\Pi_k^\star(1)^T, \Pi_k^\star(2)^T, \ldots, \Pi_k^\star(N)^T)^T$$

denotes the variables broadcast by the central entity in the last iteration at time k.

IV. A NETWORK OF RESIDENTIAL ENERGY SYSTEMS

In this section we recall a network of $\mathcal{I} \in \mathbb{N}$ residential energy systems (RESs) satisfying the dynamics (1). The network of RESs is then used in Section V to illustrate the numerical performance of Algorithm 1 and 2.

A. A network of RESs

Following [11] and the extension in [12], we assume that the system dynamics of the i-th RES, $i \in \mathbb{N}_{\mathcal{I}}$, is defined as

$$x_i(k+1) = \alpha_i x_i(k) + T\left(\beta_i u_i^+(k) + u_i^-(k)\right) \quad (16)$$
$$z_i(k) = u_i^+(k) + \gamma_i u_i^-(k) + w_i(k) \quad (17)$$

with $(\alpha_i, \beta_i, \gamma_i) \in [0,1]^3$. Thus, for the system dynamics defined in (1), we obtain the matrices $A_i(k) = \alpha_i$, $B_i(k) = [T\beta_i, \ T]$, $E_i(k) = 0$ and $F_i(k) = [1, \ \gamma_i]$ for all $k \in \mathbb{N}$ and for all $i \in \mathbb{N}_{\mathcal{I}}$.

The variables $z_i(k)$ [kW], represent the power demand drawn from/supplied to the grid at a point of common coupling by each RES and, as described by (17), depend on the net consumptions $w_i(k)$ [kW]; i.e., local power generation (e.g., solar photovoltaic panels) minus the energy demand of the residence supplemented by the use of a battery or some other appropriate storage element. The dynamics of the battery, (16), are defined through the state of charge $x_i(k)$ in [kWh] and the input variables $u_i^+(k)$ and $u_i^-(k)$ in [kW]. The input $u_i^+(k)$ represents the power drawn from the grid to increase the state of charge of the battery and the power demand $z_i(k)$, while $u_i^-(k)$ can be used to discharge the battery and decrease the power demand of the i-th RES. The parameter α_i models losses due to self-discharge, β_i and γ_i model losses due to energy conversion, and T (in hours [h]) represents the discretization in time. Additionally, the storage device of each RES is constrained by:

$$0 \leq x_i(k) \leq C_i \quad (18a)$$
$$\underline{u}_i \leq u_i^-(k) \leq 0 \quad (18b)$$
$$0 \leq u_i^+(k) \leq \overline{u}_i \quad (18c)$$
$$0 \leq \frac{u_i^-(k)}{\underline{u}_i} + \frac{u_i^+(k)}{\overline{u}_i} \leq 1 \quad (18d)$$

which define the sets $\mathbb{U}_i \subset \mathbb{R}^2$ and $\mathbb{X}_i \subset \mathbb{R}$.

If $\max\{\beta_i, \gamma_i\} < 1$ holds, energy can be *wasted*, meaning that an RES can dissipate an energy surplus by merely charging and discharging its battery. Note that this possibility is limited due to the battery constraint (18d). Additionally, it is possible to change from charging to discharging (and vice versa) between consecutive time steps k and $k+1$. Since in this case only a fraction of the time interval of length T is used for charging and discharging, the upper and lower bounds for charging and discharging have to be adapted, which

is captured by the constraint (18d). The dynamics (16)–(17) and the constraints (18) extend the models introduced in [13], [11]. Similar models of energy networks are described in [14] and [15].

B. Numerical setting

For the numerical simulations in the next section we use a dataset of $\mathcal{I} = 300$ net consumption profiles $(w_i(k))_{k \in \mathbb{N}}$, $i \in \mathbb{N}_{300}$ provided by the Australian electricity distribution company Ausgrid. (See [16] for a detailed analysis of the dataset.)

For the constraints we consider two different settings depending on the application. Either we use a set of small batteries with $C_i = 2 + \Delta C_i$[kWh] and $-\underline{u}_i + \Delta \underline{u}_i = \overline{u}_i + \Delta \overline{u}_i = 0.5$[kW] or a set of bigger batteries with $C_i = 4 + \Delta C_i$[kWh] and $-\underline{u}_i + \Delta \underline{u}_i = \overline{u}_i + \Delta \overline{u}_i = 0.75$[kW] for $i \in \mathbb{N}_{300}$. Here, for $i \in \mathbb{N}_{300}$, $\Delta C_i \in [-1, 1]$ and $\Delta \underline{u}_i, \Delta \overline{u}_i \in [-0.25, 0.25]$, or $\Delta C_i \in [-2, 2]$ and $\Delta \underline{u}_i, \Delta \overline{u}_i \in [-0.375, 0.375]$ for the bigger batteries, define random variables which distinguish the batteries of the individual RESs. Moreover, we use the constants $(\alpha_i, \beta_i, \gamma_i) = (0.99, 0.95, 0.95)$ and $x_i^0 = 0.5 \cdot C_i$[kWh] for $i \in \mathbb{N}_{300}$, and $T = 0.5$[h]. The prediction horizon N is set to $N = 48$; i.e., we solve the optimal control problem for a 24 hour horizon. The parameter ρ in the ADMM formulation is set to $\rho = 0.01$. In the visualizations we plot $\overline{u}(k) = \overline{u}^+(k) + \overline{u}^-(k)$ instead of $\overline{u}^+(k)$ and $\overline{u}^-(k)$ separately.

V. APPLICATIONS TO POWER NETWORKS

In this section we discuss several applications of the distributed predictive control Algorithm 2 using the hierarchical distributed optimization Algorithm 1 at every time step. In particular we show how the functions g and h and the set $\mathbb{S}$ can be chosen to obtain closed loop controllers with a desired performance. We here concentrate on the performance of the overall system dynamic $\overline{\Sigma}$ and not on the performance of an individual system Σ_i; i.e., we set $f_i \equiv 0$ for all $i = 1, \ldots, \mathcal{I}$, and focus on the functions g and h. However, the results and numerical simulations can be extended to the case where the functions f_i are not equal to zero.

A. Smoothing power demand

To minimize the fluctuation in the aggregated power demand we penalize the deviation from the average power demand in the prediction horizon. Thus, we define the average power demand at time $k \in \mathbb{N}$ as

$$\zeta(k) = \frac{1}{\mathcal{I}N} \sum_{j=k}^{k+N-1} \sum_{i=1}^{\mathcal{I}} w_i(j) \qquad (19)$$

and the function g as

$$g(\overline{\mathbf{a}}) = \sum_{j=k}^{k+N-1} (\overline{\mathbf{a}}(j) - \zeta(k))^2 = \|\overline{\mathbf{a}} - \mathbb{1}\zeta(k)\|^2. \qquad (20)$$

Instead of penalizing the fluctuations, one may try to obtain a smooth aggregated power demand profile. In this case the function g can be defined as

$$g(\overline{\mathbf{a}}) = \sum_{j=k}^{k+N-2} (\overline{\mathbf{a}}(j+1) - \overline{\mathbf{a}}(j))^2 = \overline{\mathbf{a}}^T \mathbf{G} \overline{\mathbf{a}} \qquad (21)$$

where $\mathbf{G} \in \mathbb{R}^{N \times N}$ is given by

$$\mathbf{G} = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & 2 & -1 \\ & & & -1 & 1 \end{pmatrix}. \qquad (22)$$

In both cases, a definition of the function h is not necessary (i.e., $h \equiv 0$). The closed loop solutions for the proposed functions g and the setting provided in Section IV-B using the small set of batteries are shown in Figure 2. We observe

Fig. 2. Comparison of the uncontrolled setting (cyan) and the controlled setting using the objective function ((20); red) and ((21); blue).

that both objective functions lead to similar results in the closed loop solution, at least in the case where the size of the batteries and the charging/discharging rates are large enough to compensate for the peaks in the uncontrolled setting. Note that even in the case where the constraints on the size of the batteries and the constraints on the charging and discharging rate are never active, a constant power demand profile can not be expected due to the time varying reference value (19).

Remark V.1. *The update of the central entity in Algorithm 1 can be computed explicitly. For* (20) *the update is given by*

$$\overline{\mathbf{a}}^{\ell+1} = \frac{\rho \mathcal{I}}{2 + \rho \mathcal{I}} \left(\overline{\mathbf{z}}^{\ell+1} + \frac{1}{\rho} \overline{\lambda}^{\ell} \right)$$

and for (21) *the update is given by*

$$\overline{\mathbf{a}}^{\ell+1} = \left(\frac{2}{\rho \mathcal{I}} G + I \right)^{-1} \left(\overline{\mathbf{z}}^{\ell+1} + \frac{1}{\rho} \overline{\lambda}^{\ell} \right)$$

where $I \in \mathbb{R}^{N \times N}$ *denotes the identity matrix.*

B. Time-varying tube constraints

The function h and the set $\mathbb{S}$ can also be defined in such a way that the aggregated power demand stays within time-varying lower and upper bounds defined by the central entity. This means that the central entity can set lower and upper bounds reflecting the minimal and maximal power which can be drawn from the main grid at a certain time instant k. Based on the predicted profiles, the central entity can decide, if an additional external power plant is necessary or can be shut down, which leads to time varying tube constraints limiting the aggregated power demand. To this end, consider the bounds

$$\underline{\mathbf{c}} \leq \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \mathbf{z}_i \leq \overline{\mathbf{c}}$$

for possibly time-varying vectors $\underline{\mathbf{c}}, \overline{\mathbf{c}} \in \mathbb{R}^N$. With $g \equiv 0$, the optimization problem

$$
\begin{aligned}
(\mathbf{z}^\star, \mathbf{a}^\star) \quad & \in \underset{(\mathbf{z},\mathbf{a})}{\arg\min} \; g(\overline{\mathbf{a}}) \\
\text{s.t.} \quad & \underline{\mathbf{c}} \leq \overline{\mathbf{a}} \leq \overline{\mathbf{c}} \\
& \mathbf{z}_i \in \mathbb{D}_i, \; \mathbf{z}_i - \mathbf{a}_i = 0, \quad \forall\, i = 1,\ldots,\mathcal{I}
\end{aligned}
$$

captures this setting. To avoid infeasibility of the minimization problem, we relax the inequality constraints to

$$\underline{\mathbf{c}} - \underline{\mathbf{s}} \leq \overline{\mathbf{a}} \leq \overline{\mathbf{c}} + \overline{\mathbf{s}}$$

for $\underline{\mathbf{s}}, \overline{\mathbf{s}} \in \mathbb{R}_{\geq 0}^N$, $\mathbf{s} := (\underline{\mathbf{s}}^T, \; \overline{\mathbf{s}}^T)^T$ and include the function $h : \mathbb{R}_{\geq 0}^{2N} \to \mathbb{R}$ penalizing the deviation from $\mathbf{s} = 0$; e.g.

$$h(\mathbf{s}) = \|\mathbf{s}\|^2. \qquad (23)$$

Thus, the overall optimization problem, which can be solved using the distributed hierarchical optimization Algorithm 1 is

$$
\left.
\begin{aligned}
(\mathbf{z}^\star, \mathbf{a}^\star, \mathbf{s}^\star) \quad & \in \underset{(\mathbf{z},\mathbf{a},\mathbf{s})}{\arg\min} \; g(\overline{\mathbf{a}}) + h(\mathbf{s}) \\
\text{s.t.} \quad & (\overline{\mathbf{a}}, \mathbf{s}) \in \mathbb{S} \\
& \mathbf{z}_i \in \mathbb{D}_i, \; \mathbf{z}_i - \mathbf{a}_i = 0, \; \forall\, i = 1,\ldots,\mathcal{I}
\end{aligned}
\right\} \quad (24)
$$

with

$$\mathbb{S} = \left\{ (\overline{\mathbf{a}}, \mathbf{s}) \in \mathbb{R}^N \times \mathbb{R}_{\geq 0}^{2N} \left| \begin{pmatrix} -I \\ I \end{pmatrix} \overline{\mathbf{a}}^T - \mathbf{s}^T \leq \begin{pmatrix} -\underline{\mathbf{c}} \\ \overline{\mathbf{c}} \end{pmatrix} \right. \right\}.$$

The optimal variables $\mathbf{s}^\star$ can be used to change the bounds $\underline{\mathbf{c}}$ and $\overline{\mathbf{c}}$ at the next time step. If $\mathbf{s}^\star = 0$ holds, the bounds $-\underline{\mathbf{c}}$ and $\overline{\mathbf{c}}$ can be decreased. If $\mathbf{s}^\star \neq 0$ holds, $\mathbf{s}^\star$ can be an indicator for the central entity to decide at which time steps in the prediction horizon an increased or decreased external power generation is necessary. Instead of using the function $g \equiv 0$ the function introduced in (21) can be used to additionally smooth the average power demand profile $\overline{\mathbf{z}}$.

In Figure 3, the closed loop solution corresponding to this setting is shown. For the simulations we use the parameters defined in Section IV-B and compare the performance of Algorithm 2 solving the optimization problem (24) with $g \equiv 0$ and $g(\overline{\mathbf{a}}) = \mathbf{a}^T G \mathbf{a}$ at every time step. The time-varying bounds change between 0.2[kWh] and 0.6[kWh] for the minimal demand and 0.4[kWh] and 0.8[kWh] for the maximal demand.

We observe that the battery size is too small to keep controlled demand profile within the bounds for all time steps. In the case where additionally the function $g(\overline{\mathbf{a}}) = \mathbf{a}^T G \mathbf{a}$ is
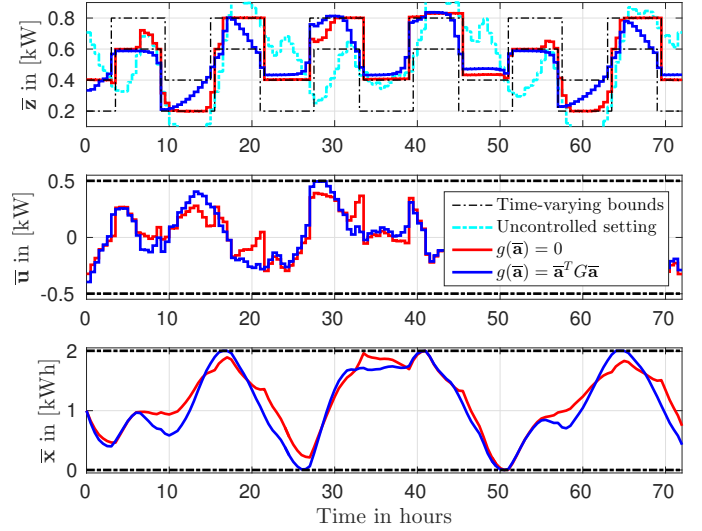

Fig. 3. Visualization of the closed loop solution for time-varying tube constraints. The graph on top shows the average power demand $\overline{z}$. The corresponding states $\overline{x}$ and the inputs $\overline{u} = \overline{u}^+ + \overline{u}^-$ are shown in the lower and in the middle graphic.

used in the objective function, the violations occur at more time steps. These violations could be decreased by decreasing the weight of the function g in the overall objective function K. As expected, the variations in $|\overline{z}(k) - \overline{z}(k-1)|$ are reduced by using $g(\overline{\mathbf{a}}) = \mathbf{a}^T G \mathbf{a}$ instead of the function $g \equiv 0$.

Again, we point out that only the central entity has to change its minimization problem at every time step. The RESs have no information about the tube constraints and do not need to adapt their respective minimization problems.
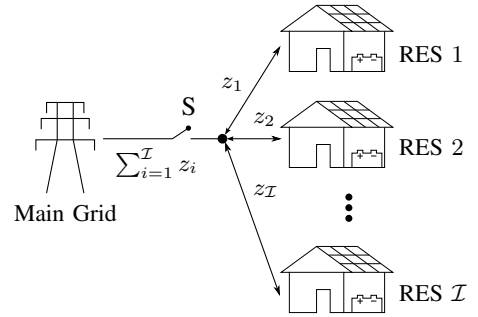
C. Islanding microgrids

Fig. 4. Visualization of the power grid. The RESs can be disconnected from the main grid after the point of common coupling using the switch S.

In this section, we define an optimization problem which serves two purposes: when solved once at time k (using Algorithm 1), its solution tells us the number of time steps $\overline{k} \in \mathbb{N}_0$ the grid can be operated in *islanded mode* after a given time instant $k + k^\star$, $k^\star \in \mathbb{N}_0$ (i.e., the microgrid can be disconnected from the main grid at time $k + k^\star$, visualized in Figure 4 using the switch S, without failing to meet the local energy demand). When solved iteratively within Algorithm 2, it yields the control strategy for keeping the microgrid in islanded mode from $k + k^\star$ to $k + k^\star + \overline{k}$.

We have the following two distinct applications in mind.

- A scheduled disconnection from the grid for an *a priori* specified time window $k^\star \geq 1$.
- An unscheduled disconnection at $k^\star = 0$.

The main difference between the two cases is that in the first scenario the microgrid can specifically prepare itself in advance by charging the batteries until time $k + k^\star$, neglecting other (possibly conflicting) objectives. Despite these differences, both settings can be handled with the proposed methodology by appropriately adapting the objective function and the constraints and writing the optimal control problem in Algorithm 1 as a convex optimization problem, demonstrating the flexibility of Algorithms 1 and 2.

The possibility of disconnecting the grid in $k^\star$ time steps is equivalent to the existence of $\mathbf{z}_i(k) \in \mathbb{D}_i$ (for $i \in \mathbb{N}_\mathcal{I}$) such that $\frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} z_i(k + k^\star) \leq 0$ is satisfied; i.e., the overall power demand is less or equal to zero at time $k + k^\star$. To find the maximal consecutive number of time steps from $k + k^\star$ to $k + k^\star + q^\star$ ($q^\star \geq 0$) such that $\frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} z_i(k + k^\star + q) \leq 0$ holds for all $q \in \{0, \ldots, q^\star\}$ we define the following minimization problem.

Problem V.2. *At a fixed time k for a given time index $k^\star \in \{0, \ldots, N-1\}$, set $M = N - k^\star$ and define the grid disconnection problem as*

$$(\bar{\mathbf{a}}^\star, \mathbf{s}^\star) \in \underset{(\bar{\mathbf{a}}, \mathbf{s}) \in \mathbb{S}}{\operatorname{argmin}} \ h(\mathbf{s})$$

where

$$\mathbb{S} = \left\{ (\bar{\mathbf{a}}, \mathbf{s}) \in \mathbb{R}^{N+M} \ \middle| \ \begin{array}{l} \begin{pmatrix} 0 & I \end{pmatrix} \bar{\mathbf{a}} - \mathbf{s} \leq 0 \\ \bar{\mathbf{a}} = \frac{1}{\mathcal{I}} \sum_{i=1}^{\mathcal{I}} \mathbf{z}_i \\ \mathbf{s} \in \mathbb{R}_{\geq 0}^M, \ \mathbf{z}_i \in \mathbb{D}_i \end{array} \right\}. \quad (25)$$

and the objective function $h : \mathbb{R}_{\geq 0}^M \to \mathbb{R}$ is defined as

$$h(\mathbf{s}) = \sum_{q=1}^{M} (M + 1 - q)^\kappa \cdot \mathbf{s}_q \quad (26)$$

for a positive constant $\kappa > 0$.

We will show that the number of leading zeros of a possibly non-unique optimal solution $\mathbf{s}^\star$ provides the maximal disconnection time if the weighting factor $\kappa > 0$ is chosen appropriately. Observe that the objective function h is linear and places a heavier penalty on the smaller indices of $\mathbf{s}$.

To give an illustrative motivation for the choice of objective function and κ, we assume that, for all $i \in \mathbb{N}_\mathcal{I}$, $\alpha_i = 1$ before we prove the general case in Theorem V.3.

Since the weighting parameters in the objective function h are positive, the constraint $\mathbf{s} \in \mathbb{R}_{\geq 0}^M$ implies $\bar{\mathbf{a}}(k^\star - 1 + q) = \mathbf{s}_q$ for all optimal $\mathbf{s}^\star$ with $\bar{\mathbf{a}}(k^\star - 1 + q) \geq 0$, $q \in \{1, \ldots, M\}$. For simplicity, consider an isolated (power) exchange between $\mathbf{s}_{q_1}$ and $\mathbf{s}_{q_2}$ ($q_1 < q_2$) of a feasible solution $\mathbf{s}$. Due to the linear system dynamics, reducing $\mathbf{s}_{q_1}$ by $\gamma_i \varepsilon$ leads to an increase of $\mathbf{s}_{q_2}$ by $\varepsilon \beta_i^{-1}$ in the case that losses have maximal impact. This is for example the case if

$$\bar{\mathbf{a}}(k^\star - 1 + q_1) = \mathbf{s}_{q_1} \quad \text{and} \quad \bar{\mathbf{a}}(k^\star - 1 + q_2) = \mathbf{s}_{q_2}$$

and $\bar{\mathbf{a}}(k^\star - 1 + q_1)$ can only be decreased by using a smaller $u_i^-(k + k^\star - 1 + q_1)$ by at least one RES i (i.e., discharge ε more from the battery at time $k^\star - 1 + q_1$) and simultaneously increase $u_i^+(k + k^\star - 1 + q_2)$ (i.e., charge ε more at time $k + k^\star - 1 + q_2$). Charging more at time $k + k^\star - 1 + q_2$ could for example be necessary to prevent violating the battery constraints (18a) at times $k + k^\star - 1 + q$ for $q > q_2$. Moreover, these considerations show that decreasing $\mathbf{s}_{q_1}$ by $\gamma_i \varepsilon$ can always be compensated by maximally increasing $\mathbf{s}_{q_2}$ by ε / β_i.

If this power exchange results in a new feasible $\tilde{\mathbf{s}}$ with

$$\tilde{\mathbf{s}}_{q_1} = \mathbf{s}_{q_1} - \gamma_i \varepsilon \quad \text{and} \quad \tilde{\mathbf{s}}_{q_2} = \mathbf{s}_{q_2} + \varepsilon / \beta_i$$

and $\tilde{\mathbf{s}}_q = \mathbf{s}_q$ for all $q \notin \{q_1, q_2\}$, and since $q_1 < q_2$ by assumption, κ has to be chosen such that the value of the objective function decreases; i.e., $h(\tilde{\mathbf{s}}) < h(\mathbf{s})$ holds. Hence we obtain the estimate

$$\begin{aligned} 0 &> h(\tilde{\mathbf{s}}) - h(\mathbf{s}) \\ &= -(M + 1 - q_1)^\kappa \gamma_i \varepsilon + (M + 1 - q_2)^\kappa \varepsilon / \beta_i \end{aligned} \quad (27)$$

or equivalently

$$\gamma_i \cdot \beta_i > \left(\frac{M + 1 - q_2}{M + 1 - q_1} \right)^\kappa.$$

Since this inequality has to hold for arbitrary $q_1 < q_2$, $q_1, q_2 \in \{1, \ldots, M\}$ the estimate

$$\gamma_i \cdot \beta_i > \left(\frac{M - 1}{M} \right)^\kappa > \left(\frac{M + 1 - q_2}{M + 1 - q_1} \right)^\kappa$$

has to be satisfied which leads to the condition

$$\kappa > \frac{\log(\gamma \cdot \beta)}{\log\left(\frac{M-1}{M} \right)}.$$

That this condition indeed suffices is rigorously shown in the following theorem.

Theorem V.3. *For an arbitrary optimal solution $(\bar{\mathbf{a}}^\star, \mathbf{s}^\star) \in \mathbb{S}$ of Problem V.2 and an arbitrary feasible solution $(\bar{\mathbf{a}}^\sharp, \mathbf{s}^\sharp) \in \mathbb{S}$, we define $q^\star, q^\sharp \in \{1, \ldots, M+1\}$ as the maximal indices such that, for all $q < q^\star$, $\mathbf{s}_q^\star = 0$ and, for all $q < q^\sharp$, $\mathbf{s}_q^\sharp = 0$. Let $\beta := \min_{i=1,\ldots,\mathcal{I}}\{\beta_i\}$ and $\gamma := \min_{i=1,\ldots,\mathcal{I}}\{\gamma_i\}$ define the maximal losses of the battery models.*

If κ is chosen such that

$$\kappa > \log\left(\beta \cdot \gamma \right) \Big/ \log\left(\frac{M-1}{M} \right) \quad (28)$$

then $q^\star \geq q^\sharp$ holds; i.e., the grid can be disconnected for at most $q^\star - 1$ time steps.

Proof. Assume that $\alpha_i = 1$ and let $(\bar{\mathbf{a}}^\star, \mathbf{s}^\star) \in \mathbb{S}$ be an optimal solution of the minimization problem with κ chosen according to Equation (28). Let $q^\star$ denote the first entry of $\mathbf{s}^\star$ which is unequal to zero, i.e., we have $\mathbf{s}_q^\star = 0$ for all $q < q^\star$ and $\mathbf{s}_{q^\star}^\star > 0$. (In the case $\mathbf{s}^\star = 0$, the statement of the theorem is trivially satisfied, and hence we can assume that $q^\star \leq M$.) Assume there exists a feasible solution $(\bar{\mathbf{a}}^\sharp, \mathbf{s}^\sharp) \in \mathbb{S}$ such that $\mathbf{s}_q^\sharp = 0$ holds for all $q < q^\sharp$ and $q^\star < q^\sharp$. We will show that the existence of $(\bar{\mathbf{a}}^\sharp, \mathbf{s}^\sharp)$ contradicts the optimality of $(\bar{\mathbf{a}}^\star, \mathbf{s}^\star)$. Additionally, we assume without loss of generality that

$$\bar{\mathbf{a}}^\star(k^\star - 1 + q) = \mathbf{s}_q^\star \quad \text{and} \quad \bar{\mathbf{a}}^\sharp(k^\star - 1 + q) = \mathbf{s}_q^\sharp$$

holds for all q with $\overline{\mathbf{a}}^\star(k^\star-1+q) \geq 0$ and $\overline{\mathbf{a}}^\sharp(k^\star-1+q) \geq 0$, respectively. Since $\mathbf{s}_{q^\star}^\sharp = 0$ and $\mathbf{s}_{q^\star}^\star = \overline{\mathbf{a}}(k^\star - 1 + q^\star) > 0$ there is an index $i \in \mathbb{N}_\mathcal{I}$ such that

$$u_i^{+\star}(k + k^\star - 1 + q^\star) > 0 \ \text{ or } \ u_i^{-\star}(k + k^\star - 1 + q^\star) > \underline{u}_i,$$

i.e., one of the constraints (18b) and (18c) is not active and it is possible to decrease

$$u_i^{+\star}(k + k^\star - 1 + q^\star) \quad \text{and/or} \quad u_i^{-\star}(k + k^\star - 1 + q^\star)$$

to reduce $z_i^\star(k + k^\star - 1 + q^\star)$ and consequently also $\overline{\mathbf{a}}(k^\star - 1 + q^\star)$ and $\mathbf{s}_{q^\star}^\star$. If we reduce $u_i^{-\star}(k + k^\star - 1 + q^\star)$ by $\varepsilon > 0$, i.e.,

$$\tilde{u}_i^{-\star}(k + k^\star - 1 + q^\star) := u_i^{-\star}(k + k^\star - 1 + q^\star) - \varepsilon \quad (29)$$

then the state x_i decreases to

$$\tilde{x}_i^\star(k + k^\star - 1 + q) := x_i^{-\star}(k + k^\star - 1 + q) - \varepsilon \quad (30)$$

for all $q > q^\star$. If $x_i^{-\star}(k + k^\star + q) > 0$ for all $q > q^\star$ then $u_i^{-\star}(k + k^\star - 1 + q^\star)$ can be changed without violating the constraints (18) and the variable $\mathbf{s}_{q^\star}^\star$ can be reduced by $\gamma_i \cdot \varepsilon / \mathcal{I}$ which contradicts the optimality. The same argument applies if it is possible to change $u_i^{+\star}(k + k^\star - 1 + q^\star)$.

Hence, we can assume, that it is only possible to change

$$u_i^{-\star}(k + k^\star - 1 + q^\star) \quad (\text{or } u_i^{+\star}(k + k^\star - 1 + q^\star),$$

respectively) by simultaneously changing

$$u_i^{-\star}(k + q) \qquad (\text{or } u_i^{+\star}(k + q))$$

at a time $q < k^\star - 1 + q^\star$ or $q > k^\star - 1 + q^\star$. Note that one of these options needs to be possible due to the existence of the solution $(\overline{\mathbf{a}}^\sharp, \mathbf{s}^\sharp)$ and since the constraints of the systems are decoupled, one can concentrate on one index $i \in \mathbb{N}_\mathcal{I}$ only.

If it is possible to increase $x_i^\star(k + q)$ by increasing $u_i^{-\star}(k + q)$ or $u_i^{+\star}(k + q)$ at time steps $q < k^\star - 1 + q^\star$ without increasing $\mathbf{s}_q^\star$ for all $q < k^\star - 1 + q^\star$ (i.e., there exist time steps $q < k^\star - 1 + q^\star$ such that $\overline{\mathbf{a}}^\star(q)$ can be increased without changing $\mathbf{s}^\star$) then this strategy increases in particular $x_i^\star(k + k^\star - 1 + q^\star)$). Hence, it is possible to decrease

$$u_i^{+\star}(k + k^\star - 1 + q^\star) \quad \text{or} \quad u_i^{-\star}(k + k^\star - 1 + q^\star)$$

without violating the constraints $x_i^\star(k + q) \geq 0$ for $q > k^\star - 1 + q^\star$; i.e., $\mathbf{s}_{q^\star}^\star$ can be reduced, violating the optimality of $\mathbf{s}^\star$.

If the strategy (29) leads to $x_i^{-\star}(k + k^\star - 1 + q) < 0$ for some $q > q^\star$, again a contradiction to optimality can be derived based on the estimate (27) and the choice of κ by decreasing $\mathbf{s}_{q^\star}^\star$ and increasing $\mathbf{s}_q^\star$ for $q > q^\star$. Hence, $(\overline{\mathbf{a}}^\star, \mathbf{s}^\sharp) \in \mathbb{S}$ does not exist, which completes the proof for $\alpha_i = 1$ for all $i \in \mathbb{N}_\mathcal{I}$.

Assume that $\alpha_i < 1$, for at least one $i \in \mathbb{N}_\mathcal{I}$. The strategy of reducing $\mathbf{s}_{q^\star}$ by increasing $\overline{\mathbf{a}}^\star(q)$ for $q < q^\star$ is applicable in the same way as in the case $\alpha_i = 1$. Moreover, if $\mathbf{s}_{q^\star}$ can be decreased using the idea of Equation (30) and simultaneously increasing $\mathbf{s}_q$ for $q > q^\star$, then the amount of energy which is lost due to self discharge for $q > q^\star$ decreases (i.e., the corresponding $u_i^{+\star}(k + k^\star - 1 + q^\star)$ or $u_i^{-\star}(k + k^\star - 1 + q^\star)$ can be decreased more before $x_i^{+\star}(k + k^\star - 1 + q) - \varepsilon = 0$ becomes active) which increases the amount $s_{q^\star}$ can be reduced to. $\square$

It has been shown in Theorem V.3 that an optimal pair $(\overline{\mathbf{a}}^\star, \overline{\mathbf{s}}^\star)$ provides the maximal disconnection time if κ is chosen such that (28) holds. However, for very large M, large values of κ are required, which lead to a numerically unstable scaling of the cost function h. Nevertheless, if a maximal disconnection time can be estimated, the presented approach can be easily generalized such that κ remains reasonably sized, e.g. the maintenance work requires at most eight hours of the 24 hours within the prediction horizon N. Moreover, numerical experiments indicate that $\kappa = 1$ works well even if (28) is violated.

Remark V.4. *If no losses are considered (i.e., $\beta = \gamma = 0$) then any value $\kappa > 0$ can be used in the objective function h. For the values $M = 48$ and $\beta = \gamma = 0.95$ we obtain $\kappa > 2.41$ from (28).*

If the maximal duration of the islanded mode is a priori specified, the constraints $\begin{pmatrix} 0 & I & 0 \end{pmatrix} \overline{\mathbf{a}} - \mathbf{s} \leq 0$ can be used instead of $\begin{pmatrix} 0 & I \end{pmatrix} \overline{\mathbf{a}} - \mathbf{s} \leq 0$ to obtain a smaller value M and, hence, a smaller κ. Moreover, observe that the objective function of the grid disconnection Problem V.2 is convex and the constraints define a convex and compact set (compactness of $\mathbb{S}$ can be easily enforced). Thus the assumptions of Theorem III.1 hold and convergence of Algorithm 1 can be concluded.

Numerical simulations: For the numerical simulations we use the objective function

$$K(\mathbf{z}, \overline{\mathbf{a}}, \mathbf{s}) = \eta \cdot \sum_{j=k}^{k+k^\star-1} \left(\overline{\mathbf{a}}(j) - \zeta(k) \right)^2 + h_{k^\star}(\mathbf{s}) \quad (31)$$

for different values of η. Here, $h_{k^\star}$ is defined as (26) and the index $k^\star$ indicates the disconnection time. For $\eta > 0$ additionally the vertical fluctuations (cf. (20)) at the time index from k to $k + k^\star - 1$ are minimized. Note that the maximal islanding interval can only be guaranteed for the case $\eta = 0$.

a) Open loop solution of Algorithm 1: In Figure 5 the solution of Algorithm 1 using the objective function (31) for $\eta = 0$ and $\eta = 100$ and $\kappa = 2.5$ are visualized. Since $\kappa = 2.5$ leads to large values in the objective function we also increase the parameter ρ to $\rho = 1$. Here, the solution corresponds to a scheduled disconnection at $k^\star = 24$ (i.e., after 12 hours). For the given initial state and parameters, the grid can be disconnected for 5 hours. In the case $\eta \neq 0$, additionally the vertical grid load is minimized in the first 12 hours. Observe that at the time the microgrid must be reconnected, the average state of charge of the batteries is still at 50%. Hence, the requirement that the microgrid is reconnected is not due to a shortage of locally stored energy, but rather due to the maximal discharging rate being too small to satisfy the microgrid demand.

b) Closed loop simulation of Algorithm 2: In Figure 6 the closed loop performance of the receding horizon Algorithm 2 using the objective function (31) for varying $k^\star$, $\eta = 0$ and $\eta = 1$, and $\kappa = 1$ is visualized. The grid operator wants to disconnect the grid after 48 time steps. Hence, the peak-to-peak variation is penalized (in the case $\eta = 1$) and the disconnection time is maximized. After 24 hours the grid is
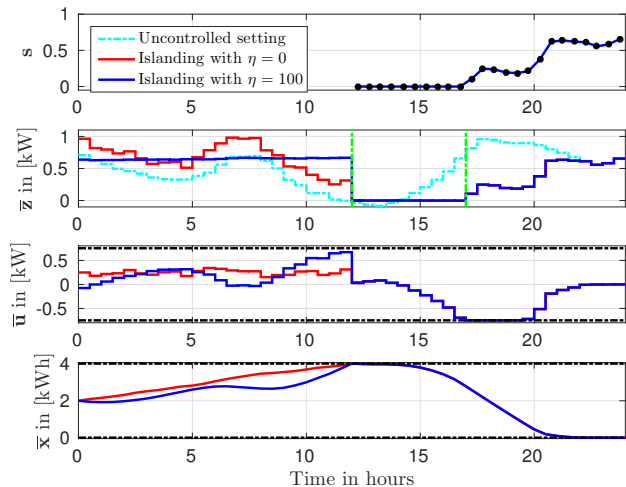
Fig. 5. Visualization of the variable **s** and the average values $\overline{x}$, $\overline{z}$, and $\overline{u} = \overline{u}^+ + \overline{u}^-$ for a single minimization problem with different weights. The microgrid is disconnected after 12 hours and can stay islanded for 5 hours. Additionally, the uncontrolled power demand without storage devices is shown for comparison.

disconnected and stays disconnected for 14 time steps, which is the maximal number of time steps. After the RESs are connected again, the simulation is continued by minimizing the deviation from the average (i.e., objective function (20)) and without an objective (i.e., $K \equiv 0$). In both cases (i.e., $\eta = 0$ and $\eta = 1$) the same islanding window is obtained. For $\eta = 1$ the energy provider additionally benefits from a flat demand profile. Even though we use $\kappa = 1$, the maximal islanding time is returned by the algorithm, which shows, that condition (28) is conservative in our application.

As already pointed out, since only the cost function of the grid operator changes in this process, the RESs do not need to change anything on the local level.
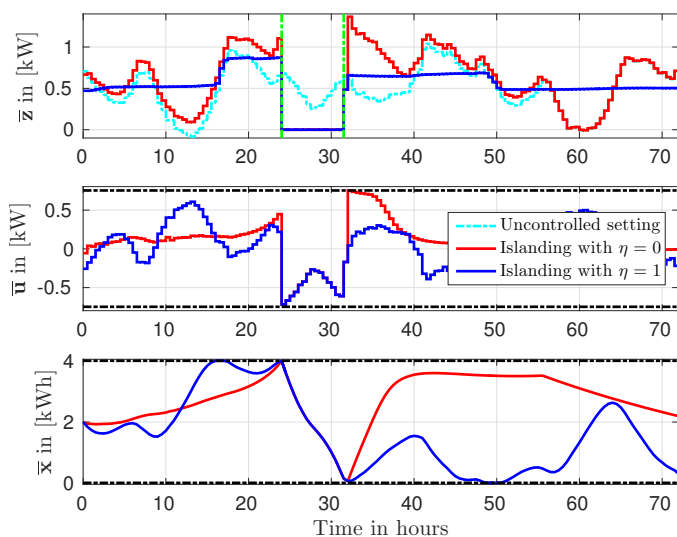

Fig. 6. Visualization of the average values $\overline{x}$, $\overline{z}$, and $\overline{u}$ for a closed loop simulation of 3 days. The maximal islanding time after 24 hours is maximized once with additionally penalizing the vertical fluctuations and once without additional penalties.

VI. CONCLUSIONS

In this paper we presented a hierarchical distributed optimization algorithm based on the alternating direction method of multipliers and designed for the predictive control of linear systems coupled in the objective function. We demonstrated how the flexibility in the distributed optimization algorithm can be used to design objective functions such that desired control objectives in the context of the optimal operation of a microgrid are satisfied. For these settings, the theoretical results were illustrated by numerical simulations.

The theoretical convergence results of the distributed optimization algorithm are based on the assumption that the feasible set of the underlying optimization problem is convex. Thus, linearity of the system dynamics is necessary for convergence of the proposed scheme. It remains an open question whether or not this approach can be generalized to the consideration of nonlinear system dynamics.

REFERENCES

[1] J. N. Tsitsiklis. *Problems in Decentralized Decision Making and Computation*. PhD thesis, MIT, Cambridge, MA, USA, 1984.
[2] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, MA, USA, 1989.
[3] D. Jakovetić, J. Xavier, and J. M. F. Moura. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59(5):1131–1146, 2014.
[4] S. S. Kia, J. Cortés, and S. Martínez. Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication. *Automatica*, 55:254–264, 2015.
[5] P. Braun, L. Grüne, C. M. Kellett, S. R. Weller, and K. Worthmann. A distributed optimization algorithm for the predictive control of smart grids. *IEEE Transactions on Automatic Control*, 2016. DOI:10.1109/TAC.2016.2525808.
[6] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.
[7] K. Worthmann, C. M. Kellett, P. Braun, L. Grüne, and S. R. Weller. Distributed and decentralized control of residential energy systems incorporating battery storage. *IEEE Transactions on Smart Grid*, 6(4):1914–1923, 2015.
[8] H.A. Rahman, M.S. Majid, A.R. Jordehi, C.K. Gan, M.Y. Hassan, and S.O. Fadhl. Operation and control strategies of integrated distributed energy resources: A review. *Renewable and Sustainable Energy Reviews*, 51:1412–1420, 2015.
[9] J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009.
[10] L. Grüne and J. Pannek. *Nonlinear Model Predictive Control. Theory and Algorithms*. Springer London, 2017.
[11] K. Worthmann, C. M. Kellett, L. Grüne, and S. R. Weller. Distributed control of residential energy systems using a market maker. In *Proc. of the 19th IFAC World Congress*, pages 11641–11646, 2014.
[12] P. Braun, T. Faulwasser, L. Grüne, C. M. Kellett, S. R. Weller, and K. Worthmann. Maximal islanding time for microgrids via distributed predictive control. In *Proc. of 22nd International Symposium on Mathematical Theory of Networks and Systems*, pages 652–659, 2016.
[13] E. L. Ratnam, S. R. Weller, and C. M. Kellett. An optimization-based approach for assessing the benefits of residential battery storage in conjunction with solar PV. In *Proc. of the IREP Symp. Bulk Power System Dynamics and Control–IX*, pages 1–8, 2013.
[14] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. Demand-side management via distributed energy generation and storage optimization. *IEEE Transactions on Smart Grid*, 4(2):866–876, 2013.
[15] M. Kraning, E. Chu, J. Lavaei, and Stephen S. P. Boyd. Dynamic network energy management via proximal message passing. *Foundations and Trends in Optimization*, 1(2):70–122, 2013.
[16] E. L. Ratnam, S. R. Weller, C. M. Kellett, and A. T. Murray. Residential load and rooftop PV generation: An Australian distribution network dataset. *International Journal of Sustainable Energy*, pages 1–20, 2015. DOI:10.1080/14786451.2015.1100196.