Resilient Information-Exchange Protocol for Distributed Model Predictive Control Schemes

Wicak Ananduta, Julian Barreiro-Gomez, Carlos Ocampo-Martinez, *Senior Member, IEEE* and Nicanor Quijano, *Senior Member, IEEE*

***Abstract*— Distributed Model Predictive Control (DMPC) strategies require local controllers to share information among each other. Considering the importance of communication in such control strategies and the failures that may occur in the information-sharing network, this paper proposes to apply the distributed consensus algorithm as an information-exchange protocol for DMPC controllers. The advantage of the proposed protocol is twofold. First, it relaxes some communication assumptions usually made for DMPC controllers. Second, under some assumptions, it provides resilience against some communication failures such that the performance and the features of the implemented distributed controller are preserved. A case study of a microgrid system is provided as an example in which some simulations are carried out to illustrate the aforementioned advantages.**

***Index Terms*— Distributed Model Predictive Control (DMPC), distributed consensus algorithm, communication failures**

I. Introduction

Model Predictive Control (MPC) is an optimization-based control framework that is able to provide stability and/or reference tracking while achieving some desired performance. Moreover, an advantage of employing such control method is that physical and/or operational constraints can be imposed directly when computing the control inputs. However, when dealing with a large-scale system (LSS), the optimization problem behind the MPC controllers can be intractable. Furthermore, considering a centralized control approach for some LSSs results in having more disadvantages in terms of scalability, flexibility, and reliability [1]. Therefore, distributed approaches, in which there exist local controllers that are assigned to the sub-systems, are promising solutions for systems of such type [1]. In a distributed MPC (DMPC) scheme, each local controller solves a smaller and tractable optimization problem by using not only local information, but also information from at least its neighbors with which they are physically coupled [2].

The key feature of any DMPC controller is that each local controller must be able to exchange some information in order to compute its control inputs [1], [2]. This feature distinguishes such controllers with the decentralized counterpart in which there is no interaction among local controllers, being suitable for sub-systems with weak dynamical coupling. It has been proved that DMPC controllers have an advantage in particular for LSSs in which the sub-systems are not weakly coupled [1]. With this feature, distributed control methods not only provide stability guarantees and recursive feasibility for such systems but also better performance than the decentralized ones [1].

Regarding the communication procedure, DMPC controllers can be classified into two categories: the approaches that require local controllers to coordinate with their neighbors either in a parallel or sequential fashion; and the distributed approaches that require each local controller to be fully connected with all the others [1]. Furthermore, depending on the algorithm, non-iterative DMPC controllers require the information to be exchanged once, whereas iterative DMPC controllers must exchange information multiple times within one sampling time [2].

The aforesaid requirements must be met during the control design stage. Nonetheless, they may not be satisfied throughout the control operation due to some issues, such as delays, data packet dropouts or disorder, and link failures, that may occur in the information-sharing network [3]. In this work, the last issue is specifically addressed. A communication failure is defined as follows: there exists at least one link of the information-sharing network that is broken. This problem may cause that the distributed control strategy cannot be performed as desired/appropriately. Failures may also result in sub-optimality of the solution or instability [3].

Communication failures in distributed control are quite relevant with the current development of a large variety of control problems in smart cities. For instance, in energy systems, the emergence of distributed energy resources and the microgrid concept inherently require a distributed control strategy. A microgrid is a group of interconnected loads and distributed energy resources that operates as a single controllable entity with respect to the grid, which can be isolated or connected to the grid [4]. Recent literature (e.g., [5], and [6]) shows that a group of microgrids is suitable to be controlled with distributed strategies. The fact that such systems are important infrastructure and that they can

W. Ananduta and C. Ocampo-Martinez are with the Automatic Control Department, Universitat Politècnica de Catalunya, Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain (emails: {wananduta, cocampo}@iri.upc.edu)

J. Barreiro-Gomez is with Learning & Game Theory Laboratory, New York University Abu Dhabi (NYUAD), Saadiyat Campus PO BOX 129188, United Arab Emirates (email: j.barreiro@nyu.edu)

Nicanor Quijano is with Departamento de Ingeniería Eléctrica y Electrónica, Universidad de los Andes, Bogotá, Colombia (email: nquijano@uniandes.edu.co)

This work has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 675318 (INCITE). The authors also thank projects DEOCS (Ref. DPI2016-76493-C3-3-R) and SGR C/marca 2015-2018. J. Barreiro-Gomez gratefully acknowledges support from U.S. Air Force Office of Scientific Research under grant number FA9550-17-1-0259.

cover a wide area brings the necessity of having distributed controllers that are not prone to communication failures.

There is some literature that answers the problem of communication failures in DMPC controllers. For instance, the work in [7] restricts the trajectory of control inputs to be inside a tube around the last transmitted trajectory. Moreover, the authors of [8] propose an observer to supplement a robust DMPC controller in order to deal with communication loss. Furthermore, a topology of the information-sharing network in which a distributed control is resilient against certain degree of failures is proposed in [9]. In addition, there are also some distributed control strategies that provide a contingency plan when a communication failure occurs. For instance, the work in [9], [10] proposes that each sub-system considers the control inputs from the disconnected neighbors to be null during the failures. The previous contributions specifically improve certain distributed control strategies in dealing with communication failures. However, to the best of our knowledge, a methodology to solve this issue that can be applied to any DMPC controller has not been discussed. Therefore, a possible methodology that copes with this problem is developed in this paper.

As the main contribution, the consensus protocol [11], [12] is proposed to be applied as an information-exchange subroutine of DMPC controllers. It can be seen as a complementary plug-in for already existing DMPC controllers in order to relax some communication requirements and mitigate the problems of communication failures. Nevertheless, the advantages of this method come with the cost of extra computation and communication, in terms of the amount of data exchanged. Finally, it is also shown how this methodology is applied to the power allocation problem of microgrids.

The remainder of the paper is organized as follows. In Section II, both the DMPC controller for LSSs and its communication issues are introduced. In Section III, the information-exchange methodology based on consensus is described. Section IV provides an example of its application to electrical grids as well as the numerical results that display the advantages of the proposed method. Finally, Section V concludes the paper.

Notation: The diagonal matrix of the column vector $\boldsymbol{x}$ and the block-diagonal matrix of the matrix A are denoted by $\mathrm{diag}(\boldsymbol{x})$ and $\mathrm{diag}(A)$. Discrete-time instant is denoted by sub-index k while continuous-time instant is denoted by sub-index t. Real numbers are denoted by $\mathbb{R}$ while integers are denoted by $\mathbb{Z}$. Moreover, $\mathbb{R}_{\geq a}$ denotes all real numbers in the set $\{i | i \geq a,\ i, a \in \mathbb{R}\}$ and similarly $\mathbb{Z}_{\geq a}$ denotes all integers in the set $\{i | i \geq a,\ i, a \in \mathbb{Z}\}$. In addition, $|\cdot|$ is the cardinality and $\|\cdot\|_2$ is the Euclidean norm. Finally, $\mathbf{0}_n$ denotes $[0\ 0\ \cdots\ 0]^\top \in \mathbb{R}^n$ and $\mathbb{1}_n = [1\ 1\ \cdots\ 1]^\top \in \mathbb{R}^n$.

II. Distributed MPC for Large-Scale Systems

Consider a discrete linear time-invariant LSS, which can be described by an undirected graph denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The LSS is composed of n number of sub-systems that are denoted by the set of the vertices $\mathcal{V} = \{1, 2, \ldots, n\}$. The set of the links, which is denoted by $\mathcal{E} \subseteq \{(i,j) | i, j \in \mathcal{V}\}$, represents whether or not there are physical connections (dynamical coupling and/or coupled constraints) among sub-systems, i.e., $(i,j) \in \mathcal{E}$ shows that the i^{th} sub-system is physically coupled to the j^{th} sub-system. The graph-representation is considered to be undirected since the interest is to the existence of communication between sub-systems, as in [7]. Therefore, for $(i,j) \in \mathcal{E}$, one cannot tell whether the i^{th} sub-system influences the j^{th} sub-system, vice versa, or both influence each other. Moreover, let $\mathcal{N}_i$ be the set of neighbors of the i^{th} sub-system, i.e., $\mathcal{N}_i = \{j | (i,j) \in \mathcal{E}\}$. The dynamics of the whole LSS are represented by the state-space model of each sub-system as follows:

$$\boldsymbol{x}_{i,k+1} = A_{ii}\boldsymbol{x}_{i,k} + B_{ii}\boldsymbol{u}_{i,k} + F_i\boldsymbol{d}_{i,k} + \boldsymbol{v}_{i,k},\ \forall i \in \mathcal{V}, \tag{1a}$$

$$\boldsymbol{v}_{i,k} = \sum_{j \in \mathcal{N}_i} \left(A_{ij}\boldsymbol{x}_{j,k} + B_{ij}\boldsymbol{u}_{j,k}\right),\ \forall i \in \mathcal{V}, \tag{1b}$$

where $\boldsymbol{x}_i \in \mathbb{R}^{n_{x,i}}$, $\boldsymbol{u}_i \in \mathbb{R}^{n_{u,i}}$, $\boldsymbol{d}_i \in \mathbb{R}^{n_{d,i}}$ and $\boldsymbol{v}_i \in \mathbb{R}^{n_{x,i}}$ denote the system states, the control inputs, the disturbances, and the existing coupling of the i^{th} sub-system with other sub-systems at time instant $k \in \mathbb{Z}_{\geq 0}$, respectively. Moreover, $A_{ij} \in \mathbb{R}^{n_{x,i} \times n_{x,j}}$, $B_{ij} \in \mathbb{R}^{n_{x,i} \times n_{u,j}}$, $F_i \in \mathbb{R}^{n_{x,i} \times n_{d,i}}$, for all $j \in \mathcal{N}_i \cup \{i\}$ and $i \in \mathcal{V}$, are the matrices that define the state-space model in (1). In addition, $n_{x,i}, n_{u,i}, n_{d,i} \in \mathbb{Z}_{\geq 1}$, for all $i \in \mathcal{V}$. Notice that (1) can be rewritten in a compacted form as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{F}\boldsymbol{d}_k, \tag{2}$$

where $\boldsymbol{x}_k = [\boldsymbol{x}_{i,k}^\top]_{i \in \mathcal{V}}^\top \in \mathbb{R}^{n_x}$, $n_x = \sum_{i \in \mathcal{V}} n_{x,i}$, $\boldsymbol{u}_k = [\boldsymbol{u}_{i,k}^\top]_{i \in \mathcal{V}}^\top \in \mathbb{R}^{n_u}$, $n_u = \sum_{i \in \mathcal{V}} n_{u,i}$, $\boldsymbol{d}_k = [\boldsymbol{d}_{i,k}^\top]_{i \in \mathcal{V}}^\top \in \mathbb{R}^{n_d}$, $n_d = \sum_{i \in \mathcal{V}} n_{d,i}$, $\boldsymbol{A} = [A_{ij}] \in \mathbb{R}^{n_x \times n_x}$, $\boldsymbol{B} = [B_{ij}] \in \mathbb{R}^{n_x \times n_u}$, and $\boldsymbol{F} = \mathrm{diag}(F_i) \in \mathbb{R}^{n_x \times n_d}$. Furthermore, the system is constrained by

$$g(\boldsymbol{x}_k, \boldsymbol{u}_k) \leq \mathbf{0}_{n_g}, \tag{3}$$

where $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_g}$ and $n_g \in \mathbb{Z}_{\geq 1}$. Notice that the function g can represent both local and global coupled constraints among the system states and/or the control inputs.

A DMPC controller is considered to be applied to the LSS. In general, according to (2) and (3), the following optimization problem, assuming the existence of feasible solutions, should be solved at each time instant k:

$$\underset{\boldsymbol{u}_{k|k}, \ldots, \boldsymbol{u}_{k+h_p-1|k}}{\text{minimize}} \quad \sum_{\ell=k}^{k+h_p-1} f(\boldsymbol{x}_{\ell|k}, \boldsymbol{u}_{\ell|k}) + f^t(\boldsymbol{x}_{k+h_p|k}) \tag{4a}$$

$$\text{subject to } \boldsymbol{x}_{\ell+1|k} = \boldsymbol{A}\boldsymbol{x}_{\ell|k} + \boldsymbol{B}\boldsymbol{u}_{\ell|k} + \boldsymbol{F}\boldsymbol{d}_{\ell}, \tag{4b}$$

$$g(\boldsymbol{x}_{\ell|k}, \boldsymbol{u}_{\ell|k}) \leq \mathbf{0}_{n_g}, \tag{4c}$$

$$\boldsymbol{x}_{k+h_p|k} \in \mathcal{X}^t, \tag{4d}$$

for all $\ell \in \{k, \ldots, k+h_p-1\}$, where the prediction horizon is denoted by $h_p \in \mathbb{Z}_{\geq 1}$. The function $f(\boldsymbol{x}_{\ell|k}, \boldsymbol{u}_{\ell|k}) : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ denotes the stage cost function, the function $f^t(\boldsymbol{x}_{k+h_p|k}) : \mathbb{R}^{n_x} \to \mathbb{R}$ corresponds to the terminal cost, and $\mathcal{X}^t \subseteq \mathbb{R}^{n_x}$ is the terminal set of the system states. Note that the terminal cost f^t and set $\mathcal{X}^t$ are introduced and defined such that the stability of the closed-loop system is guaranteed [1]. The main philosophy of a DMPC controller is to assign smaller sub-problems, derived from (4), to local controllers that belong to each sub-system. Moreover, when solving the corresponding optimization problem, each local controller must exchange some information with the others.

The exchange of information among the local controllers occurs throughout the information-sharing network. Since this network has bi-directional information flow, it is described as an undirected connected graph $\tilde{\mathcal{G}} = (\mathcal{V}, \tilde{\mathcal{E}})$. The set $\tilde{\mathcal{E}}$ shows how the local controllers share information in this network. Most of the distributed control methods require that local controllers communicate with their physical neighbors with which there is coupling, i.e., $\tilde{\mathcal{E}} = \mathcal{E}$, as in [13] and [14]. Meanwhile, other methods, e.g., the work in [15], require that local controllers exchange information with all the others. This means that $\tilde{\mathcal{G}}$ should be a complete graph. In Section III, it is shown that the communication requirements might be relaxed and an information-exchange method is proposed.

III. Consensus-Based Information Exchange

Consider the information-sharing network $\tilde{\mathcal{G}}$. Let $p_i \in \mathbb{R}$ be the information state of the i^{th} sub-system, which represents the state of the information that is coordinated in the network $\tilde{\mathcal{G}}$ at continuous time $t \in \mathbb{R}_{\geq 0}$. The consensus protocol in which multiple sub-systems are coordinating to agree on a joint state value has the following dynamics [11], [12]:

$$\dot{p}_{i,t} = \sum_{j \in \tilde{\mathcal{N}}_i} (p_{j,t} - p_{i,t}), \quad \forall i \in \mathcal{V}, \tag{5}$$

where $\tilde{\mathcal{N}}_i$ denotes the set of neighbors of the i^{th} sub-system in the information-sharing network $\tilde{\mathcal{G}}$, i.e., $\tilde{\mathcal{N}}_i = \{j | (i,j) \in \tilde{\mathcal{E}}\}$. Notice that under protocol (5), each sub-system, $i \in \mathcal{V}$, only communicates with its neighbors, $\tilde{\mathcal{N}}_i$. Furthermore, (5) can be rewritten as $\dot{\boldsymbol{p}}_t = -L\boldsymbol{p}_t$, where $\boldsymbol{p}_t = [p_{1,t}\ p_{2,t}\ \cdots\ p_{n,t}]^\top \in \mathbb{R}^n$, and L is the Laplacian matrix of $\tilde{\mathcal{G}}$.

Assumption 1: The graph $\tilde{\mathcal{G}}$ is connected. ◇

Theorem 1 (presented in [11]): Suppose that Assumption 1 holds and the sub-systems apply the consensus protocol (5). As $t \to \infty$, $\boldsymbol{p}_t$ converges to $p_i^\star = \frac{1}{n}\sum_{j\in\mathcal{V}} p_{j,0},\ \forall i \in \mathcal{V}$, where $p_{i,0}$ is the initial condition of the information state of the i^{th} sub-system. ◇

This protocol can be used conveniently to reconstruct information. For instance, information of the i^{th} sub-system can be reconstructed in the j^{th} sub-system even though they are not neighbors in the information-sharing network $\tilde{\mathcal{G}}$, i.e., $(i,j) \notin \tilde{\mathcal{E}}$. To illustrate the application of the consensus (5) as an information-exchange protocol, let $\boldsymbol{s}_i = [s_i^1\ s_i^2\ \cdots\ s_i^{n_{s,i}}]^\top \in \mathbb{R}^{n_{s,i}}$ be the collection of data that needs to be shared by the i^{th} sub-system to all other sub-systems. Note that the superscript m in s_i^m is introduced to identify the m^{th} datum in the i^{th} sub-system. Furthermore, the vector $\boldsymbol{s}_i$ can be the control sequence obtained by solving the local optimization problem, i.e., $\{\boldsymbol{u}_{i,k|k}\ \boldsymbol{u}_{i,k+1|k}\ \ldots\ \boldsymbol{u}_{i,k+h_p-1|k}\}$ and/or the current state of the sub-system $\boldsymbol{x}_{i,k}$. Thus, all sub-systems should exchange with each other $\boldsymbol{s} = [\boldsymbol{s}_1^\top\ \boldsymbol{s}_2^\top\ \cdots\ \boldsymbol{s}_n^\top]^\top \in \mathbb{R}^{n_s}$, being $n_s = \sum_{i=1}^{n} n_{s,i}$.

Assumption 2: All sub-systems have prior knowledge of the number of sub-systems (n) and the amount of data transmitted by other sub-systems ($n_{s,i}$ for all $i \in \mathcal{V}$). ◇

Corollary 1: Suppose that Assumption 1 holds. Let $\boldsymbol{p}_{i,t} = [p_{i,t}^1\ p_{i,t}^2\ \cdots\ p_{i,t}^{n_s}]^\top \in \mathbb{R}^{n_s}$ be the information state of the i^{th} sub-system. Moreover, suppose that Assumption 2 holds such that each sub-system initializes its information state as

$$\boldsymbol{p}_{i,0} = [\boldsymbol{q}_j^\top]_{j\in\mathcal{V}}^\top \in \mathbb{R}^{n_s}, \quad \text{where}$$
$$\boldsymbol{q}_j = \begin{cases} n\boldsymbol{s}_i & \text{if } j = i, \\ \boldsymbol{0}_{n_{s,j}} & \text{otherwise,} \end{cases}$$

in which $\boldsymbol{q}_j \in \mathbb{R}^{n_{s,j}}$, for all $j \in \mathcal{V}$, are auxiliary variables that decribe the initialization of the elements of $\boldsymbol{p}_{i,t}$. Then, by applying the consensus protocol,

$$\dot{\boldsymbol{p}}_{i,t} = \sum_{j \in \tilde{\mathcal{N}}_i} (\boldsymbol{p}_{j,t} - \boldsymbol{p}_{i,t}), \quad \forall i \in \mathcal{V}, \tag{6}$$

as $t \to \infty$, $\boldsymbol{p}_{i,t}$, for all $i \in \mathcal{V}$, converge to $\boldsymbol{s}$.

Proof: Let $\boldsymbol{p}_t = [\boldsymbol{p}_{1,t}^\top\ \boldsymbol{p}_{2,t}^\top\ \cdots\ \boldsymbol{p}_{n,t}^\top]^\top \in \mathbb{R}^{n_s n}$ be the information state of the overall system. There exists a permutation matrix $\boldsymbol{\Phi} \in \mathbb{R}^{n_s n \times n_s n}$ such that $\tilde{\boldsymbol{p}}_t = \boldsymbol{\Phi}\boldsymbol{p}_t$, where $\tilde{\boldsymbol{p}}_t = [\boldsymbol{p}_t^{1\top}\ \boldsymbol{p}_t^{2\top}\ \cdots \boldsymbol{p}_t^{n_s\top}]^\top \in \mathbb{R}^{n_s n}$, in which $\boldsymbol{p}_t^j = [p_{1,t}^j\ p_{2,t}^j\ \cdots\ p_{n,t}^j]^\top \in \mathbb{R}^n$ for $j = \{1, \ldots, n_s\}$. The dynamics of $\tilde{\boldsymbol{p}}_t$ are $\dot{\tilde{\boldsymbol{p}}}_t = -\boldsymbol{L}\tilde{\boldsymbol{p}}_t$, where $\boldsymbol{L} = \text{diag}(L) \in \mathbb{R}^{n_s n \times n_s n}$. Therefore, the eigenvalues of $\boldsymbol{L}$ are the same as the eigenvalues of L with the algebraic multiplicity of n_s. Hence, the claimed statement follows from the same line of proof of Theorem 1 [11, pp. 46]. ■

Remark 1: The convergence rate of the consensus is characterized by the second smallest eigenvalue of L, denoted by $\lambda_2(L)$, which is positive for a connected graph [11]. A large $\lambda_2(L)$ implies a fast convergence rate. Furthermore, it has been observed that a dense undirected graph has relatively large $\lambda_2(L)$ [12]. ◇

Remark 2: Although consensus is asymptotically achieved, in practice, sufficiently similar information can be recovered in a finite time. ◇

There are two main advantages of adding the consensus protocol as the subroutine of information exchange in a distributed control strategy at the cost of extra computation and communication. First, the requirement regarding communication of the distributed control strategy can be relaxed. For instance, in [15] each sub-system must have communication links with all the other sub-systems, while in [13] and [14], it is necessary to have communication links between two sub-systems that are neighbors in the physical network $\mathcal{G}$, i.e., $\mathcal{N}_i \subseteq \tilde{\mathcal{N}}_i$, for all $i \in \mathcal{V}$. However, by employing the consensus protocol, the only requirement of the information-sharing network topology is Assumption 1. Therefore, even if the information-sharing network is not a complete graph, the cooperative DMPC in [15] can still be applied, or if some sub-systems do not have direct information link to some of the neighbors, i.e., it is possible that $\mathcal{N}_i \nsubseteq \tilde{\mathcal{N}_i}$, for $i \in \mathcal{V}$, the algorithms proposed in [13] and [14] can still be performed.

The second advantage is the enhancement of the resilience of distributed algorithms against some communication failures. If some links in the network fail, each sub-system is still able to obtain the necessary information when using consensus as the information-exchange protocol and as long as Assumption 1 still holds. Therefore, the distributed control algorithm can still be performed under some degree of communication failures, implying that the features and performances of the controlled closed-loop system are preserved.

The main concerns of applying the consensus-based protocol are extra computation and communication, in the sense of data size, that are required to reach consensus. In this respect and in terms of practical application, some assumptions should be met. First, the available time to apply the consensus-based protocol and solve the optimization problem must be less than the sampling time of the controlled scheme. Especially for iterative-based DMPC controllers, one must ensure that the sampling time is enough to perform the DMPC algorithm and the consensus-based communication iteratively. In practice, it depends on the system complexity, i.e., the instrumentation, the hardware, and the software of the controllers. Furthermore, each sub-system should provide data storage as many as the total information data that are exchanged within the network, i.e., n_s. In addition, each sub-system should also agree on the labeling and ordering of the sub-systems. In order to optimize the data storage required, smaller connected information-sharing networks might be used so that each agent does not need to gather the information of all other agents, but only a subset of $\mathcal{V}$.

Additionally, [12] has also discussed an extension of the consensus protocol when dealing with time delays. The protocol becomes $\dot{\boldsymbol{p}}_{i,t} = \sum_{j \in \tilde{\mathcal{N}_i}} (\boldsymbol{p}_{j,(t-\tau_{i,j})} - \boldsymbol{p}_{i,(t-\tau_{i,j})}), \quad \forall i \in \mathcal{V}$, where $\tau_{i,j}$ denotes the time delay of the edge $(i,j) \in \tilde{\mathcal{E}}$.

In Section IV, the implementation of the proposed method is presented. Specifically, it is applied to a dual-decomposition-based DMPC controller that is designed to solve a power allocation problem of electrical grids.

IV. DMPC With the Consensus-Based Approach in Microgrids

Consider a power allocation problem of an electrical power network with distributed generators. In this problem setup, the set-points of the distributed generators are optimized such that the power demands and some operational constraints are met. Moreover, it is assumed that the obtained set-points can be satisfied by the local low-level controller of the generators. In addition, the problem is defined such that there exist global optimal solutions and the DMPC controller that is designed is able to compute those solutions. In this regard, one can clearly compare the performance of the controller during failures and in a normal condition.

A. Description of the Electrical Grid

The electrical power network is considered as a group of connected microgrids. A microgrid consists of local loads, a distributed generator, a battery, and a local controller. The power to satisfy the local loads can be obtained from the local distributed generator, the distributed generators of neighbors, and/or importing power from an external party.

The dynamics of the state of charge (SOC) of each battery $x_{i,k}$ are described by a discrete-time state-space model, i.e.,

$$x_{i,k+1} = a_i x_{i,k} + b_i p^{\mathrm{b}}_{i,k}, \quad \forall i \in \mathcal{V}, \tag{7}$$

where $0 < a_i < 1$ denotes the efficiency of the battery. The term $p^{\mathrm{b}}_{i,k} \in \mathbb{R}$ is the power delivered by or to the battery, and $b_i = -T_{\mathrm{s}}/e_i^{\max}$, where $T_{\mathrm{s}} \in \mathbb{R}_{\geq 0}$ is the sampling time and $e_i^{\max}$ denotes the maximum energy that can be stored. The SOC of the battery and the charging/discharging rate are bounded, i.e.,

$$x_i^{\min} \leq x_{i,k} \leq x_i^{\max}, \quad \forall i \in \mathcal{V}, \tag{8}$$

$$-p_i^{\mathrm{ch}} \leq p^{\mathrm{b}}_{i,k} \leq p_i^{\mathrm{dc}}, \quad \forall i \in \mathcal{V}, \tag{9}$$

where $x_i^{\min}, x_i^{\max} \in \mathbb{R}_{\geq 0}$ denote the minimum and the maximum SOC of the battery of the i^{th} microgrid, respectively. Note that $0 \leq x_i^{\min} \leq x_i^{\max} \leq 1$. Moreover, $p_i^{\mathrm{ch}}, p_i^{\mathrm{dc}} \in \mathbb{R}_{\geq 0}$ denote the maximum charging and discharging power.

The operational constraints for the power generated by the distributed generators $p^{\mathrm{g}}_{i,k} \in \mathbb{R}$ are the capacity and the power ramping constraints of the generators as follows:

$$p_i^{\mathrm{g,min}} \leq p^{\mathrm{g}}_{i,k} \leq p_i^{\mathrm{g,max}}, \quad \forall i \in \mathcal{V}, \tag{10}$$

$$-p_i^{\mathrm{g,do}} \leq p^{\mathrm{g}}_{i,k} - p^{\mathrm{g}}_{i,k-1} \leq p_i^{\mathrm{g,up}}, \quad \forall i \in \mathcal{V}, \tag{11}$$

where $p_i^{\mathrm{g,min}}, p_i^{\mathrm{g,max}} \in \mathbb{R}_{\geq 0}$ denote the minimum and the maximum power generated by the distributed generator of the i^{th} microgrid, respectively; while $p_i^{\mathrm{g,do}}, p_i^{\mathrm{g,up}} \in \mathbb{R}_{\geq 0}$ denote the maximum ramping down and up of the power. Furthermore, the power balance equations, which show that the demand is satisfied at each time instant, are

$$p^{\mathrm{d}}_{i,k} - p^{\mathrm{b}}_{i,k} - p^{\mathrm{g}}_{i,k} - p^{\mathrm{im}}_{i,k} - \sum_{j \in \mathcal{N}_i} p^{\mathrm{t}}_{ij,k} = 0, \quad \forall i \in \mathcal{V}, \tag{12}$$

where $p^{\mathrm{d}}_{i,k} \in \mathbb{R}_{\geq 0}$ denotes the local demand (load) of the i^{th} microgrid, $p^{\mathrm{im}}_{i,k} \in \mathbb{R}_{\geq 0}$ is the imported power from an external party, and $p^{\mathrm{t}}_{ij,k} \in \mathbb{R}$ is the power that is transferred between the i^{th} microgrid and its neighborhood $\mathcal{N}_i$. Note that there are additional coupled constraints related to $p^{\mathrm{t}}_{ij,k}$ that should be satisfied at each time instant, which are

$$p^{\mathrm{t}}_{ij,k} + p^{\mathrm{t}}_{ji,k} = 0, \quad \forall (i,j) \in \mathcal{E}. \tag{13}$$

Additionally, $p^{\mathrm{t}}_{ij,k}$ and $p^{\mathrm{im}}_{i,k}$ are limited by

$$-p_{ij}^{\mathrm{t,max}} \leq p^{\mathrm{t}}_{ij,k} \leq p_{ij}^{\mathrm{t,max}}, \quad \forall (i,j) \in \mathcal{E}, \tag{14}$$

$$p^{\mathrm{im}}_{i,k} \leq p^{\mathrm{im,max}}, \quad \forall i \in \mathcal{V}, \tag{15}$$

where $p_{ij}^{\mathrm{t,max}}, p^{\mathrm{im,max}} \in \mathbb{R}_{\geq 0}$ are the upper bounds.

By introducing the vector $\boldsymbol{u}_{i,k} = [\boldsymbol{u}^{l\top}_{i,k}\ \boldsymbol{u}^{c\top}_{i,k}]^\top$ with $\boldsymbol{u}^{l}_{i,k} = [p^{\mathrm{b}}_{i,k}\ p^{\mathrm{g}}_{i,k}\ p^{\mathrm{im}}_{i,k}]^\top$ and $\boldsymbol{u}^{c}_{i,k} = [p^{\mathrm{t}}_{ij,k}]$, for all $j \in \mathcal{N}_i$, and considering a quadratic economical cost as the stage cost function, i.e., $f_{i,k} = \boldsymbol{u}^\top_{i,k} R_i \boldsymbol{u}_{i,k}$, where $R_i \in \mathbb{R}^{|\boldsymbol{u}_{i,k}| \times |\boldsymbol{u}_{i,k}|}$, for all $i \in \mathcal{V}$, are positive definite matrices, the centralized optimization problem that needs to be solved in an MPC scheme can be written as

$$\begin{aligned} \underset{\{\{\boldsymbol{u}_{i,\ell|k}\}_{i\in\mathcal{V}}\}_{\ell=k}^{k+h_p-1}}{\text{minimize}} \quad & \sum_{\ell=k}^{k+h_p-1}\sum_{i=1}^{n} f_{i,\ell}(\boldsymbol{u}_{i,\ell|k}) && \text{(16a)}\\ \text{subject to} \quad & x_{i,\ell+1|k} = a_i x_{i,\ell|k} + \boldsymbol{b}_i \boldsymbol{u}_{i,\ell|k}, && \text{(16b)}\\ & E_i \boldsymbol{u}_{i,\ell|k} + \boldsymbol{e}_{i,\ell} \leq \boldsymbol{0}, && \text{(16c)}\\ & \boldsymbol{u}^c_{i,\ell|k} + \sum_{j\in\mathcal{N}_i} G_{ij}\boldsymbol{u}^c_{j,\ell|k} = \boldsymbol{0}, && \text{(16d)} \end{aligned}$$

for all $i \in \mathcal{V}$ and $\ell \in \{k, \dots, k + h_p - 1\}$. The dynamics of the SOC of the batteries are rewritten as in (16b) in which $\boldsymbol{b}_i = [b_i \quad \boldsymbol{0}_{2+|\mathcal{N}_i|}]$. The constraints that only include local decision, which are (8)-(12), and (14)-(15), are compacted in (16c) with the appropriate E_i and $\boldsymbol{e}_{i,\ell}$. Moreover, coupled constraints in (13) are rewritten as in (16d).

B. DMPC based on Dual Decomposition

A DMPC controller that is based on dual decomposition is considered. In this approach, Problem (16) is dualized and decomposed into smaller problems. Then, a distributed gradient-based algorithm, which requires information sharing, is developed by considering the obtained decomposition [16]. Since Problem (16) is convex, the solutions of the distributed algorithm converge to the global optimal solutions [16]. Denoting $\boldsymbol{\lambda}_{i,\ell} \in \mathbb{R}^{|\mathcal{N}_i|}$ as the Lagrange multipliers associated to the coupled constraints (16d) and $\boldsymbol{\psi}_{i,\ell} = \left(\boldsymbol{u}^c_{i,\ell|k} + \sum_{j\in\mathcal{N}_i} G_{ij}\boldsymbol{u}^c_{j,\ell|k}\right)$, for all $\ell \in \{k, \dots, k+h_p-1\}$ and all $i \in \mathcal{V}$, the iterative procedure that solves the dual problem associated to (16) is obtained and shown in Algorithm 1. In this algorithm, each sub-system should solve the local optimization problem in (17) and update its Lagrange multipliers via the gradient-ascent method at each iteration. Moreover, it is worth to emphasize that there are two steps of exchanging information between one agent and its neighbors at each iteration (step 4 and 6). The *default protocol* of exchanging information is that a sub-system exchanges immediately the necessary information with the neighbors through the direct communication links available among them. On the other hand, if the consensus-based protocol is applied, it is assumed that the information shared at the end of the information-exchange steps is similar enough such that it does not affect the convergence of the algorithm. Additionally, the iterations stop when $r_{\max}$, which denotes the maximum number of iterations available in one sampling time, is reached.

C. Simulation Results and Discussion

Numerical simulations are carried out in MATLAB. The topology of the grid and the information-sharing network of the controllers in these simulations are depicted in Figure 1. Moreover, it is considered that each microgrid has at least two neighbors in the information-sharing network in order to maintain the network connectivity in the scenarios of communication failure satisfying Assumption 1. Additionally, some links of the information-sharing network are broken in some simulations (indicated by double lines in Figure 1).

Algorithm 1 The Dual-Ascent DMPC Algorithm

```
1: All sub-systems, $i = \{1, 2, \dots, n\}$ do
2: Set $r = 1$ and initialize $\boldsymbol{\lambda}^{(r)}_{i,\ell}$.
3: while $r \leq r_{\max}$ do
4:     Information exchange: Receive $\boldsymbol{\lambda}^{(r)}_{j,\ell}$, $\forall \ell \in \{k, \dots, k+h_p-1\}$, from the neighbors, all $j \in \mathcal{N}_i$, and send $\boldsymbol{\lambda}^{(r)}_{i,\ell}$, $\forall \ell \in \{k, \dots, k+h_p-1\}$, to the neighbors.
5:     Solve the local optimization problem:
           $\underset{\{\boldsymbol{u}_{i,\ell|k}\}_{\ell=k}^{k+h_p-1}}{\text{minimize}} \sum_{\ell=k}^{k+h_p-1} \left(f_{i,\ell}(\boldsymbol{u}_{i,\ell|k}) + \boldsymbol{y}^\top_{i,\ell}\boldsymbol{u}^c_{i,\ell|k}\right)$   (17)
           subject to (16b) and (16c),
       for all $\ell \in \{k, \dots, k+h_p-1\}$, where $\boldsymbol{y}^\top_{i,\ell} = \boldsymbol{\lambda}^{(r)\top}_{i,\ell} + \sum_{j\in\mathcal{N}_i} \boldsymbol{\lambda}^{(r)\top}_{j,\ell} G_{ji}$.
6:     Information exchange: Receive $\boldsymbol{u}^c_{j,\ell|k}$, $\forall \ell \in \{k, \dots, k+h_p-1\}$, from the neighbors and send $\boldsymbol{u}^c_{i,\ell|k}$, $\forall \ell \in \{k, \cdots, k+h_p-1\}$, to the neighbors.
7:     Update $\boldsymbol{\lambda}_{i,\ell}$ for all $\ell \in \{k, \dots, k+h_p-1\}$ as $\boldsymbol{\lambda}^{(r+1)}_{i,\ell} = \boldsymbol{\lambda}^{(r)}_{i,\ell} + \gamma\boldsymbol{\psi}_{i,\ell}$, for $0 < \gamma < 1$.
8: end while
```

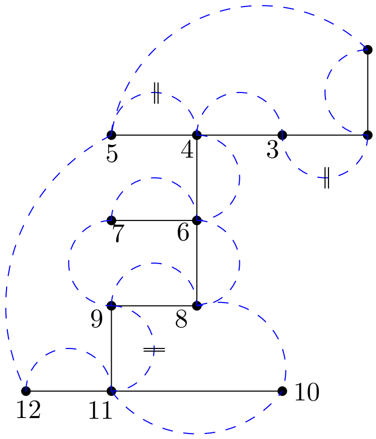

Fig. 1. The microgrids (dots) in their electrical network $\mathcal{G}$ (solid lines) and information-sharing network $\tilde{\mathcal{G}}$ (dashed lines).The double lines, ||, indicate the broken links during communication failures in the simulations.

The sampling time of the system is 15 minutes, and the simulation time is one day (96 steps). Moreover, $h_p = 4$ and the parameters corresponding to each microgrid are shown in Table I. Note that the weight matrix in the cost function is defined as $R_i = \text{diag}\left([c^{\text{b}}_i \; c^{\text{g}}_i \; c^{\text{im}}_i \; c^{\text{t}}_i \mathbb{1}^\top_{|\mathcal{N}_i|}]^\top\right)$. Furthermore, it is assumed that each local controller knows the load trajectory at each time instant.

In order to compare the performance of the proposed method, four scenarios are simulated (see Table II). During the failures, the communication links $\{(2,3),(4,5),(9,11)\} \in \tilde{\mathcal{E}}$ are broken (see Figure 1) at $k = \{4, 10-13, 30, 31, 50, 80-82\}$. Notice that the choice of the broken links is motivated by the requirement to maintain the connectivity of the information-sharing network while disconnecting two microgrids that are neighbors in the physical network $\mathcal{G}$.

TABLE I

PARAMETERS OF THE MICROGRIDS

Parameters	Value	Unit	Sub-system (i)
$x_i^{\min}$, $x_i^{\max}$	40, 200	kWh	1,2,. . . ,12
p_i^{ch}, p_i^{dc}	50, 50	kW	1,2,. . . ,12
$p_i^{\mathrm{g,min}}$, $p_i^{\mathrm{g,max}}$	0, 1000	kW	1,2,. . . ,12
$p_i^{\mathrm{g,do}}$, $p_i^{\mathrm{g,up}}$	100, 100	kW	1,2,. . . ,12
$p_{ij}^{\mathrm{t,max}}$, $p_i^{\mathrm{im,max}}$	100, 2000	kW	1,2,. . . ,12
a_i	0.95	-	1,2,. . . ,12
c_i^{b}, c_i^{im}, c_i^{t}	0.1, 250, 0.1	-	1,2,. . . ,12
c_i^{g}	2	-	2, 5, 11
	10	-	1, 3, 4, 6, 7, 8, 9, 10, 12

TABLE II

THE AVERAGE STAGE COST AT TIME INSTANTS AT WHICH THE FAILURE OCCURS

Scenario	Protocol, Communication	Cost (Proportional)
1	Default, no failures	1.00
2	Consensus-based, no failures	1.00
3	Default, with failures	1.18
4	Consensus-based, with failures	1.00

Scenario 1 provides the benchmark performance of the DMPC controller. In this scenario, the controller produces the global optimal solution at each time instant. In Scenario 3, when the information links are broken, the sub-systems that are disconnected do not receive information from their neighbors when the default protocol is applied. Thus, the information from the neighbors is considered as null by the controllers, as in [9], [10]. It results in decisions that are in average 18% more expensive than in the optimal case. Furthermore, it is important to highlight that the proposed protocol is able to cope with communication failures (Scenario 4). It can be seen by comparing the data from Scenarios 1 and 4, where they both produce the same cost.

V. CONCLUSION AND FUTURE WORK

The standard distributed consensus protocol is exploited as an information-sharing method and it can be implemented to any DMPC controller. As an example, the implementation in a DMPC based on dual decomposition for the power allocation problem in electrical grids has been shown. As future work, it is interesting to relax the assumptions of the method such that it can work in a more general information-sharing network.

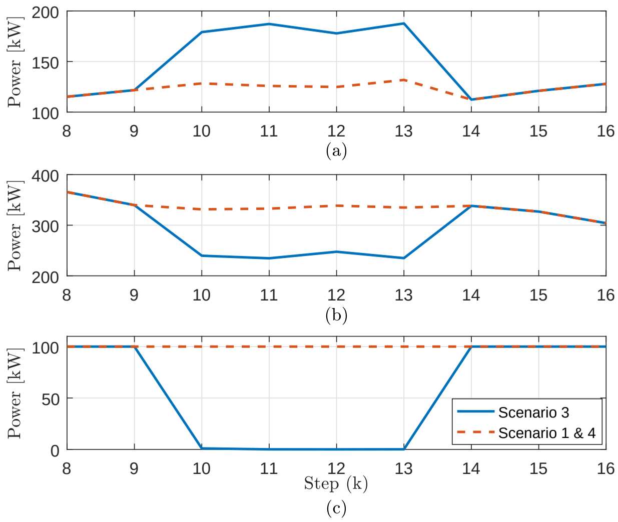

Fig. 2. The operation of sub-system 3 and 2 in Scenario 3 (solid lines) and Scenario 1 & 4 (dashed lines). (a) Power generated by sub-system 3 (p_3^{g}), (b) Power generated by sub-system 2 (p_2^{g}), and (c) Power transferred from sub-system 2 to sub-system 3 (p_{32}^{t}).

REFERENCES

[1] P. D. Christofides, R. Scattolini, D. Muñoz de la Peña, and J. Liu, "Distributed model predictive control: A tutorial review and future research directions," *Computers & Chemical Engineering*, vol. 51, pp. 21–41, 2013.

[2] R. Scattolini, "Architectures for distributed and hierarchical model predictive control – A review," *Journal of process control*, vol. 19, no. 5, pp. 723–731, 2009.

[3] X. Ge, F. Yang, and Q.-L. Han, "Distributed networked control systems: A brief overview," *Information Sciences*, vol. 380, pp. 117–131, 2017.

[4] R. H. Lasseter, "Microgrids," in *Proceedings of IEEE Power Engineering Society Winter Meeting*, vol. 1, New York, USA, 2002, pp. 305–308.

[5] W. Shi, X. Xie, C.-C. Chu, and R. Gadh, "Distributed optimal energy management in microgrids," *IEEE Transactions on Smart Grid*, vol. 6, no. 3, pp. 1137–1146, 2015.

[6] A. Pantoja and N. Quijano, "A population dynamics approach for the dispatch of distributed generators," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4559–4567, 2011.

[7] R. M. Schaich, M. A. Müller, and F. Allgöwer, "A distributed model predictive control scheme for networks with communication failure," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 12 004–12 009, 2014.

[8] D. Kumar, W. Al-Gherwi, and H. Budman, "Robust-distributed MPC tolerant to communication loss," *Computers & Chemical Engineering*, vol. 88, pp. 30–38, 2016.

[9] M. H. Nazari, S. Grijalva, and M. Egerstedt, "Distributed frequency regulation in the presence of communication failure," in *Proceedings of the IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, Minneapolis, USA, 2016, pp. 1–5.

[10] M. Heidarinejad, J. Liu, D. Muñoz de la Peña, J. F. Davis, and P. D. Christofides, "Handling communication disruptions in distributed model predictive control," *Journal of Process Control*, vol. 21, no. 1, pp. 173–181, 2011.

[11] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[12] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[13] M. Farina and R. Scattolini, "Distributed predictive control: A non-cooperative algorithm with neighbor-to-neighbor communication for linear systems," *Automatica*, vol. 48, no. 6, pp. 1088–1096, 2012.

[14] C. Conte, C. N. Jones, M. Morari, and M. N. Zeilinger, "Distributed synthesis and stability of cooperative distributed model predictive control for linear systems," *Automatica*, vol. 69, pp. 117–125, 2016.

[15] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, "Cooperative distributed model predictive control," *Systems & Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.

[16] F. Farokhi, I. Shames, and K. H. Johansson, "Distributed MPC via dual decomposition and alternative direction method of multipliers," in *Distributed model predictive control made easy*, J. M. Maestre and R. R. Negenborn, Eds. Springer, 2014, pp. 115–131.