On the Mitigation of Communication Failures in Distributed Model Predictive Control Strategies

Wicak Ananduta, Julian Barreiro-Gomez, Carlos Ocampo-Martinez, and Nicanor Quijano,

***Abstract*— Information sharing among local controllers is the key feature of any Distributed Model Predictive Control (DMPC) strategy. This paper addresses the problem of communication failures in DMPC strategies and proposes a distributed solution to cope with them. The proposal consists in an information-exchange protocol that is based on distributed projection dynamics. By applying this protocol as a complementary plug-in to a DMPC strategy, the controllers become resilient against communication failures and relax the communication requirements. Furthermore, a discussion on the selection criteria of the information-sharing network and a reconfiguration algorithm, which is a contingency procedure to maintain the connectivity of the network, are also presented. In order to demonstrate the performance and advantages when adopting the proposed approach, a case study of a power-network control problem is considered.**

***Index Terms*— Distributed MPC, information-exchange protocol, distributed projection dynamics**

I. Introduction

Research in Model Predictive Control (MPC) has been extensively done since the past decades (see e.g., [1]–[4] for a review). Such studies include the application of MPC to large-scale systems, i.e., systems that have a large number of inputs, states, and/or outputs. Since an MPC controller computes control inputs by solving an optimization problem, the problem may become intractable when the system is too large [3]. Therefore, many works in the literature have proposed non-centralized MPC approaches, in which there are multiple local controllers as opposed to the centralized approach, which has a single central controller, in order to cope with such problem [3], [4]. Furthermore, non-centralized MPC approaches could also introduce additional benefits such as better scalability, flexibility, and safety than the centralized counterpart [4].

Non-centralized MPC approaches can be classified into two broad categories: decentralized and distributed approaches. The main difference between these two schemes is the existence of communication among local controllers, which can be found in the latter category [3]. The availability of communication plays an important role for the performance of distributed MPC (DMPC) approaches, which is better than the decentralized ones when the couplings between sub-systems are not weak [4]. In fact, in certain cases, some DMPC approaches are also able to achieve similar performance as the centralized counterpart [4].

Different DMPC approaches require different communication structure as well as different ways to exchange information among local controllers [3]. Some DMPC approaches require local controllers to share information iteratively while others require that the information exchange is made only once at each time instant. In terms of information-sharing network, some approaches, such as [5]–[7], require a neighbour-to-neighbour communication while other approaches, e.g., [8], [9], require all controllers to exchange information with all the others. In any way, the information-sharing network, through which the exchange of information occurs, is important for systems that use a DMPC approach.

Some issues of an information-sharing network may arise during the operation of the associated system. Those issues include communication failures (total loss of communication links), delays, and data packet drops [10]. This paper focuses on the failures (loss of links) in the information-sharing network, in which some local controllers are no longer able to communicate with others. As mentioned in [10] and later discussed, communication failures may lead to severe problems such as the inability of the controllers to compute control inputs or the sub-optimality of the solutions.

Some recent literature has addressed the problem of communication failures in a distributed control strategy, in particular DMPC. The authors of [11] analyse the performance degradation of a DMPC strategy that is based on Nash optimality during such failures while assuming that the algorithm is convergent. The authors of [12] propose a scheme where the sub-systems assume that their neighbours take null control actions during communication failures. In [13], the authors develop a methodology to extend the DMPC strategy that is proposed in [14] such that it can cope with communication failures. The methodology involves substituting the coupled constraints with tube-based constraints that restrict the control inputs. Furthermore, [15] proposes to add an observer for a robust DMPC strategy. Hence, during a communication failure, the state bounds are estimated by the observer and are posed as extra constraints into the DMPC

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 and Game Theory Laboratory, New York University in 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)

The work of W. Ananduta has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 675318 (INCITE). J. Barreiro-Gomez acknowledges U.S. Air Force Office of Scientific Research under grant number FA9550-17-1-0259. N. Quijano acknowledges SGR C/marca project 2015-2017.

design. Additionally, a resilient information-sharing network architecture for distributed frequency regulation is proposed in [16]. Moreover, the controller in [16] adopts a *zero-bias* control strategy and allows other sub-systems to stabilize themselves.

The aforementioned contributions only improve some DMPC strategies to tackle communication failures in a way that they specifically add or modify the algorithms. Therefore, they limit the application of the solution only to the DMPC strategies that are discussed in those papers. In this regard, different from the works previously discussed, this paper proposes a communication protocol that can be applied regardless DMPC strategy that is used to control the system. The protocol is an iterative algorithm that requires local controllers to communicate at each iteration until the information that is received converges to the correct value. It is based on the distributed projection dynamics (DPD) [17], [18] and can also be perceived as a distributed consensus protocol [19]. For an extensive treatment of the notion of distributed population dynamics, the reader is referred to [17], [18].

In this paper, an application of the proposed approach is illustrated in a power-allocation problem of interconnected microgrids. These kinds of systems are of large nature and spread over large geographical areas. Furthermore, in the current development of these systems, the power generation is shifted toward distributed generation, particularly renewable energy sources [20]. For these reasons, distributed controllers have been viewed as suitable control approaches [20]–[23]. In this regard and highlighting the fact that power networks are considered to be a critical infrastructure, it is important to ensure the resiliency of the control approach against failures, in particular communication failures.

In summary, the main contribution of this paper is the information-exchange protocol that is based on DPD, as presented in Section III. This protocol can be applied to any DMPC strategy and improves the resilience of the DMPC strategy against communication failures as well as relaxes the communication requirements. Criteria to select a suitable information-sharing network for the proposed protocol and a network reconfiguration algorithm that supplements the protocol are also proposed. Moreover, prior to proposing the protocol, Section II provides an analysis of the impact of communication failures in a DMPC strategy. Furthermore, the advantages of the proposed protocol in a power allocation problem of microgrids are shown in Section IV.

Notations

The statements $A \succeq 0$ and $B \succ 0$ imply A and B are positive semi-definite and positive definite, respectively. The set of real numbers is denoted by $\mathbb{R}$ whereas the set of integers is 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 operator and $\|\cdot\|_2$ is the Euclidean norm. The vector $\mathbf{0}_n$ denotes $[0 \ 0 \ \cdots \ 0]^\top \in \mathbb{R}^n$ and, similarly, $\mathbb{1}_n = [1 \ 1 \ \cdots \ 1]^\top \in \mathbb{R}^n$. The subscript $k \in \mathbb{Z}_{\geq 0}$ denotes discrete-time instants while the subscript $t \in \mathbb{R}_{\geq 0}$ denotes continuous-time instants. For the vectors $v_j \in \mathbb{R}^{n_j}$ with $j \in \mathcal{N}$, the operator $[v_j^\top]_{j\in\mathcal{N}}^\top \in \mathbb{R}^{\sum_{j\in\mathcal{N}} n_j}$ denotes the column-wise concatenation that results in a column-wise vector. Moreover, consider a time-dependent vector $\boldsymbol{u}_{i,k}$ and a prediction horizon h_p. Then, $\tilde{\boldsymbol{u}}_{i,k}$ denotes the trajectory of $\boldsymbol{u}_{i,k}$ over the prediction horizon h_p, i.e., $\tilde{\boldsymbol{u}}_{i,k} = \begin{bmatrix} \boldsymbol{u}_{i,k|k}^\top & \boldsymbol{u}_{i,k+1|k}^\top & \cdots & \boldsymbol{u}_{i,k+h_p-1|k}^\top \end{bmatrix}^\top$. Finally, $\mathrm{rnd}(\cdot)$ is the round operator that approximates the argument to the nearest integer. In the case there are two nearest integers, the argument is approximated to the larger integer.

II. Distributed MPC and Communication Failures

In this section, a general description of the system that is considered and the DMPC approach are presented. Afterwards, a discussion of the impact of communication failures in a DMPC strategy is provided.

A. Distributed MPC

Consider a time-invariant linear large-scale system (LSS) that consists of n sub-systems. Let it be described by an undirected graph $\mathcal{G} = (\mathcal{S}, \mathcal{E})$, where the set of nodes, $\mathcal{S} = \{1, 2, \ldots, n\}$ denotes its sub-systems and the set of unweighted links, $\mathcal{E} \subseteq \mathcal{S} \times \mathcal{S}$, represents the existence of coupling between the sub-systems, i.e., $(i, j) \in \mathcal{E}$ implies that the i^{th} sub-system is physically coupled to the j^{th} sub-system. Furthermore, $\mathcal{N}_i$ denotes the set of the neighbours of the i^{th} sub-system, i.e., $\mathcal{N}_i = \{j : (i, j) \in \mathcal{E}, \forall j \neq i\}$. Let the dynamics of this LSS be described by the discrete-time state-space model of each sub-system, i.e., for each $i \in \mathcal{S}$,

$$\boldsymbol{x}_{i,k+1} = A_{ii}\boldsymbol{x}_{i,k} + B_{ii}\boldsymbol{u}_{i,k}, \tag{1}$$

where the states and the control inputs the i^{th} sub-system are denoted by the vectors $\boldsymbol{x}_i \in \mathbb{R}^{n_{x,i}}$ and $\boldsymbol{u}_i \in \mathbb{R}^{n_{u,i}}$, respectively. Moreover, A_{ii} and B_{ii} are the state-space matrices with the appropriate sizes that define the dynamics in (1). Note that the dynamics of each sub-system might also depend on the states, inputs, and/or disturbances of the neighbouring sub-systems. However, for simplicity of the exposition in Subsection II-B, such dynamical couplings are omitted. Instead, some constraints that couple the inputs will be introduced. Moreover, $\mathcal{G}$ is undirected and unweighted since the interest of this paper is in the existence of communication, as in [13].

The control inputs are constrained by local polytopic constraints described by $H_i\boldsymbol{u}_{i,k} \leq \boldsymbol{h}_i$, for each $i \in \mathcal{S}$, where $H_i \in \mathbb{R}^{n_{h,i}\times n_{u,i}}$ and $\boldsymbol{h}_i \in \mathbb{R}^{n_{h,i}}$ are constant matrix and vector, respectively. Furthermore, there also exist some hard coupled constraints on the control inputs that are described as $\sum_{i\in\mathcal{S}} G_i\boldsymbol{u}_{i,k}^c \leq \boldsymbol{g}$, where $G_i \in \mathbb{R}^{n_g\times n_{u^c,i}}$, for all $i \in \mathcal{S}$, and $\boldsymbol{g} \in \mathbb{R}^{n_g}$ are constant matrices and vector, respectively. Moreover, $\boldsymbol{u}_{i,k}^c \in \mathbb{R}^{n_{u^c,i}}$, for all $i \in \mathcal{S}$, denote the elements of input $\boldsymbol{u}_{i,k}$ that appear in the coupled constraints. In other words, $\boldsymbol{u}_{i,k}$ can be arranged such that $\boldsymbol{u}_{i,k} = [\boldsymbol{u}_{i,k}^{l\top} \ \boldsymbol{u}_{i,k}^{c\top}]^\top$, where $\boldsymbol{u}_{i,k}^l \in \mathbb{R}^{n_{u^l,i}}$ denotes the local control inputs. Notice that the coupled constraints may be constraints in which the sub-systems share a limited common input source. Hence,

the optimization problem that should be solved in an MPC scheme at each time instant $k \in \mathbb{Z}_{\geq 0}$ is

$$\begin{aligned}
\underset{\{\tilde{\boldsymbol{u}}_{i,k}\}_{i\in\mathcal{S}}}{\text{minimize}} \quad & \sum_{i\in\mathcal{S}} J_i(\boldsymbol{x}_{i,k}, \tilde{\boldsymbol{u}}_{i,k}) && \text{(2a)}\\
\text{subject to} \quad & \boldsymbol{x}_{i,\ell+1|k} = A_{ii}\boldsymbol{x}_{i,\ell|k} + B_{ii}\boldsymbol{u}_{i,\ell|k},\ \forall i \in \mathcal{S}, && \text{(2b)}\\
& H_i\boldsymbol{u}_{i,\ell|k} \leq \boldsymbol{h}_i,\ \boldsymbol{x}_{i,k+h_p|k} \in \mathcal{X}_i^t,\ \forall i \in \mathcal{S}, && \text{(2c)}\\
& \sum_{i\in\mathcal{S}} G_i\boldsymbol{u}^c_{i,\ell|k} \leq \boldsymbol{g}, && \text{(2d)}
\end{aligned}$$

for all $\ell \in \{k, \ldots, k+h_p-1\}$, where the trajectory of control input sequence throughout h_p at time instant k for each sub-system is denoted by $\tilde{\boldsymbol{u}}_{i,k}$. Notice that the cost function over h_p is separable and it is defined as

$$\begin{aligned}
J_i(\boldsymbol{x}_{i,k}, \tilde{\boldsymbol{u}}_{i,k}) = & \sum_{\ell=k}^{k+h_p-1} \boldsymbol{x}^\top_{i,\ell|k} Q_i \boldsymbol{x}_{i,\ell|k} + \boldsymbol{u}^\top_{i,\ell|k} R_i \boldsymbol{u}_{i,\ell|k}\\
& + J_i^t(\boldsymbol{x}_{i,k+h_p|k}),
\end{aligned}$$

where $Q_i \succeq 0$, $R_i \succ 0$ and it is assumed that Problem (2) is feasible. Moreover, $J_i^t(\boldsymbol{x}_{i,k+h_p})$, which denotes a convex terminal cost function and $\mathcal{X}_i^t$, which denotes a terminal set that is convex, compact, and contains the origin, are introduced and defined in order to guarantee stability [4], [6]. Therefore, since the cost function is strictly convex and by definition the constraints are also convex, Problem (2) is convex.

In a DMPC scheme, sub-problems are derived by decomposing (2) and they are solved by local controllers assigned to the sub-systems. When solving the sub-problems, the local controllers need to share some information among each other. Therefore, a DMPC strategy requires an information-sharing network. In this regard, let the information-sharing network of system $\mathcal{G}$ be described by an undirected connected graph $\tilde{\mathcal{G}} = (\mathcal{S}, \tilde{\mathcal{E}})$, where the set $\tilde{\mathcal{E}}$ shows how the local controllers share information in this network, i.e., the link $(i, j) \in \tilde{\mathcal{E}}$ infers that there exists a bidirectional information flow between the i^{th} and j^{th} sub-systems through a communication channel. Many distributed MPC methods, e.g., [5]–[7], require that local controllers share information with their neighbours in $\mathcal{G}$. This means that $\tilde{\mathcal{E}} = \mathcal{E}$. Other methods, e.g., the method presented in [8] and [9], require full information exchange, i.e., $\tilde{\mathcal{G}}$ should be a complete graph.

B. Impacts of Communication Failures

The availability of communication between sub-systems is the main feature of DMPC strategies. Therefore, this section is dedicated to show, by an example, what could happen when there are failures in the information-sharing network, i.e., at least one communication link fails for a certain time slot. During this period, the affected sub-systems cannot communicate with each other while employing a DMPC strategy. For this example, without loss of generality, consider a system that consists of two sub-systems, i.e., $\mathcal{S} = \{1, 2\}$. Moreover, consider that the system has dynamics and constraints as described in the previous subsection, implying that Problem (2) must be solved by the DMPC controllers. Furthermore, let $\tilde{\boldsymbol{u}}^\star_{i,k}$, for all $i \in \mathcal{S}$, be the optimal solution of this problem.

Now, a DMPC strategy that is based on dual decomposition is considered to be implemented and the control inputs obtained by this strategy are analysed. In principle, the DMPC strategies that are based on dual decomposition are iterative and use the concept of duality in convex optimization [24]. The iterations, which are called the distributed dual ascent, are intended to optimize the Lagrange multipliers that are associated with the coupled constraints in order to solve its dual problem [25]. The solution of this algorithm converges to the global optimal value if the primal problem is convex and the cost function is strictly convex [25]. The iteration steps of the distributed dual ascent algorithm that is applied by the i^{th} sub-system for problem (2) are as follows: 1.

1) The update of the decision $\tilde{\boldsymbol{u}}_{i,k}$ at the r^{th} iteration is

$$\begin{aligned}
\tilde{\boldsymbol{u}}^{(r)}_{i,k} = \underset{\tilde{\boldsymbol{u}}_{i,k}}{\text{argmin}}\ & J_i(\boldsymbol{x}_{i,k}, \tilde{\boldsymbol{u}}_{i,k}) + \sum_{\ell=k}^{k+h_p-1} \left(\boldsymbol{\lambda}^{(r)\top}_{i,\ell|k} G_i \boldsymbol{u}^c_{i,\ell|k}\right)\\
& \text{subject to}\ \ \text{(2b)}, \text{(2c)},\ \forall \ell \in \{k, \ldots, k+h_p-1\},
\end{aligned} \tag{3}$$

   in which $\boldsymbol{\lambda}^{(r)}_{i,\ell|k}$, for all $\ell \in \{k, \ldots, k+h_p-1\}$ and $i \in \mathcal{S}$, are the Lagrange multipliers that are associated with the input-coupled constraints (2d).
2) The update of the Lagrange multipliers is $\boldsymbol{\lambda}^{(r+1)}_{i,\ell|k} = \boldsymbol{\lambda}^{(r)}_{i,\ell|k} + \gamma\left(G_1\boldsymbol{u}^{c(r)}_{1,\ell|k} + G_2\boldsymbol{u}^{c(r)}_{2,\ell|k} - \boldsymbol{g}\right)$, for all $\ell \in \{k, \ldots, k+h_p-1\}$, where $\gamma \in \mathbb{R}$, $0 < \gamma < 1$.

Notice that all steps of the dual ascent algorithm can be done locally and each sub-system requires the information of the decision of its neighbourhood in order to update the Lagrange multipliers. When exchanging information, consider the information sharing network $\tilde{\mathcal{G}} = (\mathcal{S}, \tilde{\mathcal{E}})$, where $\tilde{\mathcal{E}} = \{(1, 2)\}$.

During a communication failure, it is assumed that each sub-system uses the information of the neighbourhood from the previous time instant and considers the unknown information to be null, as in [12] and [16]. This is precisely stated in Assumption 1.

Assumption 1: Consider that the communication link $(i, j) \in \tilde{\mathcal{E}}$ fails. During the period of the failure, the i^{th} sub-system considers the input trajectory of the the j^{th} sub-system ($\tilde{\boldsymbol{u}}_{j,k}$), denoted by $\tilde{\boldsymbol{w}}_{j,k} = \begin{bmatrix}\boldsymbol{w}^\top_{j,k} & \cdots & \boldsymbol{w}^\top_{j,k+h_p-1}\end{bmatrix}^\top$, as follows:

$$\boldsymbol{w}_{j,\ell} = \begin{cases} \boldsymbol{u}_{j,\ell|k-1}, & \ell = k, \ldots, k+h_p-2,\\ \boldsymbol{0}_{n_{u,j}}, & \ell = k+h_p-1, \end{cases}$$

and vice versa. ♢

Note that based on the structure of $\boldsymbol{u}_{i,k}$, $\boldsymbol{w}_{j,k} = [\boldsymbol{w}^{l\top}_{j,k}\ \boldsymbol{w}^{c\top}_{j,k}]^\top$, where $\boldsymbol{w}^l_{j,k}$ and $\boldsymbol{w}^c_{j,k}$ correspond to the local inputs and the coupled inputs, respectively. In this case, during a communication failure, both sub-system cannot communicate and receive the necessary information, which is the coupled inputs of their neighbour. It implies that the steps of updating the Lagrange multipliers for both sub-systems

become

$$\boldsymbol{\lambda}_{i,\ell|k}^{(r+1)} = \boldsymbol{\lambda}_{i,\ell|k}^{(r)} + \gamma \left(G_i \boldsymbol{u}_{i,\ell|k}^{c(r)} + G_j \boldsymbol{w}_{j,\ell}^{c(r)} - \boldsymbol{g} \right), \quad (4)$$

for all $\ell \in \{k, \dots, k+h_p-1\}$ and $i \in \mathcal{S}$, where $j \in \mathcal{N}_i$. The modification of the dual-ascent algorithm implies that instead of solving Problem (2), the DMPC controllers solve the following optimization problem:

$$\begin{aligned} \underset{\{\tilde{\boldsymbol{u}}_{i,k}\}_{i\in\mathcal{S}}}{\text{minimize}} \quad & \sum_{i\in\mathcal{S}} J_i(\boldsymbol{x}_{i,k}, \tilde{\boldsymbol{u}}_{i,k}) && \text{(5a)} \\ \text{subject to} \quad & \text{(2b), (2c)}, \\ & G_i \boldsymbol{u}_{i,\ell|k}^c + G_j \boldsymbol{w}_{j,\ell}^c \leq \boldsymbol{g}, \quad \forall j \in \mathcal{N}_i, && \text{(5b)} \end{aligned}$$

for all $i \in \mathcal{S}$ and $\ell \in \{k, \dots, k+h_p-1\}$. The difference between Problem (5) and Problem (2) is in the existence of coupled constrains (2d) in Problem (2), which are replaced by (5b). Notice that the inequalities (5b) are not coupled constraints since $\boldsymbol{w}_{j,\ell}^c$, for all $j \in \mathcal{S}$ and $\ell \in \{k, \dots, k+h_p-1\}$ are known information (see Assumption 1). Hence, Problem (5) is separable by definition.

An analysis regarding the performance of the controllers can be made by evaluating Problem (5). Problem (5) may be infeasible even though Problem (2) is feasible due to the changes of the constraints. In this case, the dual-ascent algorithm with the updating steps according to (3) and (4) will not obtain a solution. Suppose that Problem (5) is feasible. Since it is convex, the dual-ascent algorithm can find a solution. Consider that each sub-system obtains a solution $\tilde{\boldsymbol{u}}_{i,k}^*$, for all $i \in \mathcal{S}$. Thus, $\tilde{\boldsymbol{u}}_k^* = \left[\tilde{\boldsymbol{u}}_{1,k}^{*\top} \ \tilde{\boldsymbol{u}}_{2,k}^{*\top}\right]^\top$ must be examined, since this is the control input that will be applied to the system. The solution $\tilde{\boldsymbol{u}}_k^*$ may be an infeasible solution for Problem (2), i.e., it violates some constraints of Problem (2). The DMPC scheme in [12], which apply Assumption 1 during failures, also assumes that null control input is a feasible solution. However, the latter assumption does not always hold in general. Furthermore, supposing that $\tilde{\boldsymbol{u}}_k^*$ is feasible, if $\tilde{\boldsymbol{u}}_k^* = \left[\tilde{\boldsymbol{u}}_{1,k}^{\star\top} \ \tilde{\boldsymbol{u}}_{2,k}^{\star\top}\right]^\top$, then $\tilde{\boldsymbol{u}}_k^*$ is the optimal solution. Otherwise, $\tilde{\boldsymbol{u}}_k^*$ is a suboptimal solution. Thus, this DMPC strategy does not have a guarantee on the feasibility, let alone the stability and the optimality, during communication failures even though the problem is convex with a strictly convex cost function. Similar issues may also be found when the strategy is applied to a more complex and larger problem and in other DMPC strategies. Hence, increasing the resilience of communication infrastructure is important. Section III discusses a proposal to improve the resilience of the information-sharing network.

III. Information-Exchange Protocol based on Distributed Projection Dynamics

The proposal focuses on the methodology of sharing information, which is independent from the control strategy. Thus, it is a general and complementary methodology that can be applied to any DMPC method.

A. Description of the Protocol

Recall the information-sharing network as the graph $\tilde{\mathcal{G}} = (\mathcal{S}, \tilde{\mathcal{E}})$, where $\tilde{\mathcal{E}}$ shows how the local controllers are connected with each other in the information-sharing network. Furthermore, consider that some nodes (sub-systems) require information denoted by $\boldsymbol{s}_v = [s_v^1 \ s_v^2 \ \cdots \ s_v^{n_{s,v}}]^\top \in \mathbb{R}^{n_{s,v}}$, from the v^{th} node. Depending on the DMPC strategy applied to the system and the couplings in the system, $\boldsymbol{s}_v$ may consist of either state or input information, which is required by some or all other nodes to run the DMPC algorithm. Therefore, there exists a sub-graph $\tilde{\mathcal{G}}_v = (\mathcal{S}_v, \tilde{\mathcal{E}}_v) \subseteq \tilde{\mathcal{G}}$, where $\mathcal{S}_v \subseteq \mathcal{S}$ is the set that consists of node v and the nodes that require $\boldsymbol{s}_v$, while $\mathcal{E}_v \subseteq \mathcal{E}$ is the set of links that connect the nodes in $\mathcal{S}_v$. For instance, in the DMPC strategies that require neighbour-to-neighbour communication, some neighbours $j \in \mathcal{N}_v$ might be included in $\mathcal{G}_v$ while the other neighbours are included in the other information-sharing subgraphs. In order to apply the protocol, the following assumptions must hold.

Assumption 2: The undirected sub-graph $\tilde{\mathcal{G}}_v$ is connected. ♢

Assumption 3: The node v, which sends the information, has prior knowledge of $|\mathcal{S}_v|$, while all $i \in \mathcal{S}_v \backslash \{v\}$ know $n_{s,v}$. ♢

Note that, following Assumption 2, it is possible that there are some nodes in $\mathcal{S}_v$ that do not need the information $\boldsymbol{s}_v$, but they are required as intermediate nodes in order to ensure the connectivity of $\tilde{\mathcal{G}}_v$.

The proposed information-exchange protocol is based on the following dynamics:

$$\dot{\boldsymbol{p}}_{i,t} = \alpha \sum_{j\in\tilde{\mathcal{N}}_i} (\boldsymbol{p}_{j,t} + \boldsymbol{r}_j - \boldsymbol{p}_{i,t} - \boldsymbol{r}_i), \quad \forall i \in \mathcal{S}_v, \quad (6a)$$

$$\boldsymbol{q}_{i,t} = \boldsymbol{p}_{i,t} + \boldsymbol{r}_i, \quad \forall i \in \mathcal{S}_v, \quad (6b)$$

where $\boldsymbol{q}_i, \boldsymbol{r}_i, \boldsymbol{p}_i \in \mathbb{R}^{n_{s,v}}$ are the information state, the reference input, and the internal state of the i^{th} node, respectively, $\tilde{\mathcal{N}}_i$ is the set of nodes that are the neighbours of the i^{th} node in $\tilde{\mathcal{G}}_v$, i.e., $(i,j) \in \tilde{\mathcal{E}}_v$ for all $j \in \tilde{\mathcal{N}}_i$, and $\alpha \in \mathbb{R}_{>0}$ is a constant gain. Furthermore, the reference inputs of all nodes in $\mathcal{S}_v$ are given by

$$\boldsymbol{r}_i = \begin{cases} |\mathcal{S}_v| \boldsymbol{s}_v, & i = v, \\ \boldsymbol{0}_{n_{s,v}}, & \text{otherwise}, \end{cases} \quad (7)$$

and the internal states are initialized as

$$\boldsymbol{p}_{i,0} = \boldsymbol{0}_{n_{s,v}}, \quad \forall i \in \mathcal{S}_v. \quad (8)$$

It is shown in (7) that the information $\boldsymbol{s}_v$ is used as the reference input of the v^{th} node. In the following subsection, the convergence of the protocol is discussed.

Remark 1: The dynamics (6) are in the class of dynamic consensus, which tracks the average of the changing reference inputs. The reader is referred to [26] for further discussions on this topic. ♢

Remark 2: The vectors $\boldsymbol{q}_i, \boldsymbol{p}_i, \boldsymbol{r}_i \in \mathbb{R}^{s_v}$ for all $i \in \mathcal{S}_v$ are dedicated only for the transmission of information $\boldsymbol{s}_v$. Thus, each local controller needs to allocate a different data storage

for acquiring another information, i.e., if there are more than one source nodes. $\diamondsuit$

B. Convergence of the Protocol

Theorem 1 below states the convergence of the proposed information-exchange protocol.

Theorem 1: Suppose that Assumptions 2 and 3 hold. Under the dynamics (6) and by providing the references as in (7) and the initial value of the auxiliary state as in (8), the equilibrium point of the information state, i.e., $\boldsymbol{q}_i^\star = \boldsymbol{s}_v$, for each $i \in \mathcal{S}_v$, is asymptotically stable.

Proof: Firstly, notice that the dynamic equations (6a) correspond to distributed projection dynamics (DPD) [17] with the fitness functions $\boldsymbol{f}_{i,t} : \mathbb{R}^{n_{s,v}} \to \mathbb{R}^{n_{s,v}}$, for all $i \in \mathcal{S}_v$, which are defined as $\boldsymbol{f}_{i,t} = -(\boldsymbol{p}_{i,t} + \boldsymbol{r}_i)$. The protocol dynamics (6a) can be rewritten in the form of the DPD as

$$\dot{\boldsymbol{p}}_{i,t} = \alpha|\tilde{\mathcal{N}}_i|\boldsymbol{f}_{i,t} - \alpha \sum_{j\in\tilde{\mathcal{N}}_i} \boldsymbol{f}_{j,t}, \quad \forall i \in \mathcal{S}_v. \tag{9}$$

Note that $\boldsymbol{f}_{i,t} = [f_{i,t}^1\ f_{i,t}^2\ \cdots\ f_{i,t}^{n_{s,v}}]^\top$, where $f_{i,t}^l = -p_{i,t}^l - r_i^l$, for each $l \in \{1,\dots,n_{s,v}\}$, is related to the information datum s_v^l. Then, these dynamics can be rearranged into

$$\dot{\check{\boldsymbol{p}}}_t^l = -\alpha L(\tilde{\mathcal{G}}_v)\check{\boldsymbol{f}}_t^l, \quad \forall l \in \{1,2,\dots,n_{s,v}\}, \tag{10}$$

where $\dot{\check{\boldsymbol{p}}}_t^l = [\dot{p}_{i,t}^l]_{i\in\mathcal{S}_v}^\top$, $\check{\boldsymbol{f}}_t^l = [f_{i,t}^l]_{i\in\mathcal{S}_v}^\top$, for all $l \in \{1,\dots,n_{s,v}\}$, and $L(\tilde{\mathcal{G}}_v) \in \mathbb{R}^{|\mathcal{S}_v|\times|\mathcal{S}_v|}$ is the Laplacian of the graph $\tilde{\mathcal{G}}_v$. Hence, it is shown in (10) that the dynamics depend on the structure of the graph $\tilde{\mathcal{G}}_v$.

Secondly, it is necessary to show that the set

$$\Delta = \left\{ \boldsymbol{p}_t \in \mathbb{R}^{|\mathcal{S}_v|n_{s,v}} : \sum_{i\in\mathcal{S}_v} p_{i,t}^l = m^l,\ \forall l \in \{1,\dots,n_{s,v}\} \right\},$$

is invariant. This statement can be concluded by the fact that $\mathbb{1}_{|\mathcal{S}_v|}^\top L(\tilde{\mathcal{G}}_v) = 0$, i.e., $\sum_{i\in\mathcal{S}_v} \dot{p}_{i,t}^l = -\alpha\mathbb{1}_{|\mathcal{S}_v|}^\top L(\tilde{\mathcal{G}}_v)\check{\boldsymbol{f}}_t^l = 0$, for all $l \in \{1,\dots,n_{s,v}\}$. Furthermore, by denoting the whole auxiliary states as $\boldsymbol{p}_t = [\boldsymbol{p}_{i,t}^\top]_{i\in\mathcal{S}_v}^\top$ and the equilibrium point as $\boldsymbol{p}^\star = [\boldsymbol{p}_i^{\star\top}]_{i\in\mathcal{S}_v}^\top$, where $\boldsymbol{p}_i^\star$ is the equilibrium point of the i^{th} sub-system, and, based on Assumption 2, the equilibrium point $\boldsymbol{p}^\star$ can be characterized from (10), i.e., $\boldsymbol{p}^\star = \{\boldsymbol{p}_t \in \Delta : \dot{\check{\boldsymbol{p}}}_t^l = 0,\ \forall l \in \{1,\dots,n_{s,v}\}\} = \{\boldsymbol{p}_t \in \Delta : \check{\boldsymbol{f}}_t^l \in \text{span}\{\mathbb{1}_{|\mathcal{S}_v|}\},\ \forall l \in \{1,\dots,n_{s,v}\}\}$, since $\text{span}\{\mathbb{1}_{|\mathcal{S}_v|}\}$ is the nullspace of $L(\mathcal{G}_v)$ when $\mathcal{G}_v$ is connected.

Now the solution of $\boldsymbol{q}_i^\star$ is derived from $\boldsymbol{p}_i^\star$. Recall that Δ is invariant. Due to the initialization in (8), $\sum_{i\in\mathcal{S}_v} p_{i,t}^l = \sum_{i\in\mathcal{S}_v} p_{i,0}^l = 0$. Therefore, based on this fact and since $\check{\boldsymbol{f}}_t^l \in \text{span}\{\mathbb{1}_{\mathcal{S}_v}\}$ implies $f_i^l(p_i^{l\star}) = f_j^l(p_j^{l\star})$ for all $i,j \in \mathcal{S}_v$, it is obtained that $f_i^l(p_i^{l\star}) = -\frac{1}{|\mathcal{S}_v|}\sum_{j\in\mathcal{S}_v}(p_{j,0}^l + r_j)$, which implies $p_i^{l\star} = -r_i + \frac{1}{|\mathcal{S}_v|}\sum_{j\in\mathcal{S}_v} r_j$, for all $i \in \mathcal{S}_v$ and $l \in \{1,\ \dots\ n_{s,v}\}$. Hence, the whole vectors $\boldsymbol{p}_{i,t}$, for all $i \in \mathcal{S}_v$, have the following equilibrium point:

$$\boldsymbol{p}_i^\star = -\boldsymbol{r}_i + \frac{1}{|\mathcal{S}_v|}\sum_{j\in\mathcal{S}_v} \boldsymbol{r}_j = -\boldsymbol{r}_i + \boldsymbol{s}_v, \tag{11}$$

since according to (7) and Assumption 3, $\sum_{j\in\mathcal{S}_v} \boldsymbol{r}_j = |\mathcal{S}_v|\boldsymbol{s}_v$. Thus, by substituting $\boldsymbol{p}_{i,t}$ into (6b) with the expression in (11), it is obtained that the equilibrium points of the information states are $\boldsymbol{q}_i^\star = \boldsymbol{s}_v$, for all $i \in \mathcal{S}_v$.

Finally, the convergence of the internal state $\boldsymbol{p}_{i,t}$ is shown since it implies the convergence of $\boldsymbol{q}_{i,t}$ to the equilibrium point $\boldsymbol{q}_i^\star$. To this end, consider the radially unbounded Lyapunov function candidate: $V(\boldsymbol{p}_t) = \sum_{i\in\mathcal{S}_v} \frac{1}{2}\left(\boldsymbol{p}_{i,t}^\top\boldsymbol{p}_{i,t} - \boldsymbol{p}_i^{\star\top}\boldsymbol{p}_i^\star\right) + \boldsymbol{r}_i^\top(\boldsymbol{p}_{i,t} - \boldsymbol{p}_i^\star)$, where $V(\boldsymbol{p}_t) \to \infty$ as $\|\boldsymbol{p}_t\| \to \infty$. The function $V(\boldsymbol{p}^\star) = 0$ and $V(\boldsymbol{p}_t) > 0$ for all $\boldsymbol{p}_t \neq \boldsymbol{p}^\star$. The time derivative of $V(\boldsymbol{p}_t)$, i.e.,

$$\begin{aligned}
\dot{V}(\boldsymbol{p}_t) &= \sum_{i\in\mathcal{S}_v} \boldsymbol{f}_{i,t}^\top\dot{\boldsymbol{p}}_{i,t} \\
&= \sum_{i\in\mathcal{S}_v}\left(-\alpha\boldsymbol{f}_{i,t}^\top\left(|\tilde{\mathcal{N}}_i|\boldsymbol{f}_{i,t} - \sum_{j\in\tilde{\mathcal{N}}_i}\boldsymbol{f}_{j,t}\right)\right) \\
&= \sum_{l=1}^{n_{s,v}}\left(-\alpha\sum_{i\in\mathcal{S}_v}\left(f_{i,t}^l|\tilde{\mathcal{N}}_i|f_{i,t}^l - f_{i,t}^l\sum_{j\in\tilde{\mathcal{N}}_i} f_{j,t}^l\right)\right) \\
&= \sum_{l=1}^{n_{s,v}}\left(-\alpha\check{\boldsymbol{f}}_t^{l\top}L(\mathcal{G}_v)\check{\boldsymbol{f}}_t^l\right).
\end{aligned}$$

Since $\tilde{\mathcal{G}}_v$ is connected by assumption, $L(\tilde{\mathcal{G}}_v) \succeq 0$. Hence, $\dot{V}(\boldsymbol{p}_t) \leq 0$, for all $\boldsymbol{p}_t \in \mathbb{R}^{|\mathcal{S}_v|n_{s,v}}$. Furthermore, notice that equality $\dot{V}(\boldsymbol{p}_t) = 0$ only holds when $\check{\boldsymbol{f}}_t^l \in \text{span}\{\mathbb{1}_{|\mathcal{S}_v|}\}$, for all $l \in \{1,\dots,n_{s,v}\}$, i.e., at the equilibrium point $\boldsymbol{p}^\star$. Then, applying the LaSalle-invariance principle, convergence to the equilibrium point $\boldsymbol{p}^\star$ is concluded [27]. Hence, $\boldsymbol{q}_i^\star = \boldsymbol{s}_v$, for all $i \in \mathcal{S}_v$, are also asymptotically stable. ■

Remark 3: The convergence rate of the protocol depends on the structure of the network. It can be seen from the representation of the protocol in (10)*, which is similar to the standard consensus algorithm formulation. The convergence rate of such protocol is indicated by the second smallest eigenvalue of the Laplacian of the graph [28].* $\diamondsuit$

Remark 4: Although the information states asymptotically converge to $\boldsymbol{q}_i^\star = \boldsymbol{s}_v$, for all $i \in \mathcal{S}$, in practice, sufficiently similar information can be recovered in a finite time. $\diamondsuit$

C. Advantages of the Protocol

Two main advantages of the DPD-based information-exchange protocol are highlighted as follows. Firstly, it relaxes some assumptions that are required by most of DMPC strategies. Particularly, it relaxes the requirement of the information-sharing network topology. As discussed in Section II-A, different DMPC strategies might have different requirements regarding the information-sharing network topology supposing that the information is exchanged directly as required. However, by employing the proposed protocol, these requirements are relaxed such that $\tilde{\mathcal{G}}$ does not have to fulfil certain topological structure. Instead, only the connectivity of $\tilde{\mathcal{G}}$ is necessary (Assumption 2). Secondly, this protocol also enhances the resiliency of DMPC-type controllers against communication failures. According to Assumption 2, the information can still be exchanged although

some links of the network $\tilde{\mathcal{G}}$ fail as long as the network is still connected. Therefore, to some extent of link failures, a DMPC strategy that uses the proposed protocol to exchange information can still be performed.

The advantages provided by the protocol also come with some costs, which are extra computation and communication, in terms of the amount of data that is exchanged. This is due to the fact that all sub-systems should reach consensus by iteratively exchanging information and applying (6) before obtaining the correct information from their neighbours. Therefore, one must ensure that the total time to exchange information using the DPD-based protocol and to compute the control inputs is smaller than the sampling time of the controlled system. In practice, the satisfaction of this assumption depends on the system complexity, i.e., the instrumentation and the other hardware as well as the software, e.g., the optimization solver.

D. Selecting the Information-Sharing Graphs

Now consider that the system has an information-sharing network that connects its sub-systems in a certain way. In this subsection, a discussion on how to choose the information-sharing sub-graph, $\tilde{\mathcal{G}}_v$, from the available information-sharing network is provided. Two criteria of selection are the resiliency of the network against communication failures and the convergence rate of the proposed method. It is a direct implication that, when the information-sharing graph has more links (edges) connecting the sub-systems, the chance that the graph is still connected when a failure occurs is higher. Furthermore, as stated in Remark 3, the second smallest eigenvalue of $L(\tilde{G}_v)$ indicates the convergence rate of the protocol in the sense that a larger eigenvalue implies a faster convergence rate. As noted in [28], the second smallest eigenvalue of a sparse Laplacian is relatively small compared to a dense Laplacian. Hence, these criteria lead to the fact that the network should have as many links as possible.

Redundancy is required when dealing with communication failures. Thus, when selecting an information-sharing graph, it is important to consider having redundant links in the information-sharing graph. For instance, a path is not a suitable structure since once a link is disconnected, the graph is disconnected. In that sense, a cyclic graph is more redundant because the proposed method could still be applied when one link of this graph is broken. Furthermore, it is obvious that a complete graph is the most suitable one. On the other hand, the topology of an LSS usually has a sparse Laplacian matrix due to the fact that a sub-system is usually only coupled with other closest sub-systems. This may imply its information-sharing network has sparse Laplacian as well. However, this network can be decomposed into some sub-graphs that do not have sparse Laplacian, which implies they may have faster convergence rate. By also considering the redundancy criterion, then one may be able to use a smaller yet redundant information-sharing graph.

As an example, consider the information-sharing graph depicted in Fig. 1. Its Laplacian has the second smallest eigenvalue of 0.23. However, now consider its sub-graphs

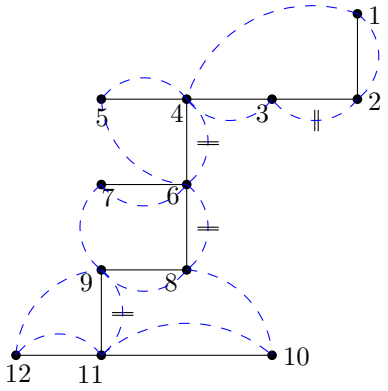

Fig. 1. A system with 12 nodes. The physical network $\mathcal{G}$ and the information-sharing network $\tilde{\mathcal{G}}$ are represented by the solid and dashed lines, respectively.

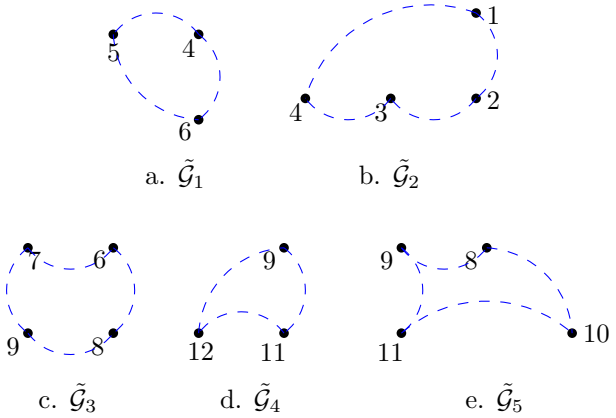

Fig. 2. Decomposition of $\tilde{\mathcal{G}}$ into 5 smaller connected sub-graphs.

that are depicted in Fig. 2. Notice that the sub-graphs that are formed are cyclic, in order to satisfy the redundancy requirement. Among these sub-graphs, the smallest value of the Laplacian second smallest eigenvalues is 2.00. This means that the convergence rate of the proposed method is much faster by using the smaller sub-graphs as the information-sharing graphs among the sub-systems. However, the redundancy of the sub-graphs is not as good as the overall graph, $\tilde{\mathcal{G}}$. For instance, consider sub-graph $\tilde{\mathcal{G}}_3$ (Fig. 2c) and suppose that the links $(7, 9)$ and $(6, 7)$ are broken, then this sub-graph is not connected anymore, which implies the nodes in this sub-graph (6, 7, 8, and 9) cannot exchange information among each other. However, the graph $\tilde{\mathcal{G}}$ is in fact still connected, allowing the protocol to be applied and all nodes to exchange information when the $\tilde{\mathcal{G}}$ is used as the information-sharing graph of all information.

E. Reconfiguration of Information-Sharing Graphs

In this section, a distributed algorithm to reconfigure the information-sharing graphs is proposed. The reconfiguration is a contingency procedure when the information-sharing graphs are disconnected due to the failures, i.e., Assumption 2 does not hold. Moreover, the reconfiguration can also be applied even though the graphs are still connected in order to maintain redundancy.

Consider again the information-sharing network $\tilde{\mathcal{G}} = (\mathcal{S}, \tilde{\mathcal{E}})$ and its sub-graph $\tilde{\mathcal{G}}_v = (\mathcal{S}_v, \tilde{\mathcal{E}}_v) \subseteq \tilde{\mathcal{G}}$ that is used to share information of the v^{th} node, denoted by $\boldsymbol{s}_v$, among the sub-systems in $\mathcal{S}_v \subseteq \mathcal{S}$. Now, denote the set of links that fail by $\tilde{\mathcal{E}}_f \subset \tilde{\mathcal{E}}_v$. Moreover, let the following Assumption 4 holds.

Assumption 4: Each node $i \in \mathcal{S}$ has prior knowledge of its neighbours in the information-sharing network $\tilde{\mathcal{G}}$, i.e., $\tilde{\mathcal{N}}_i$ and $\tilde{\mathcal{M}}_i := \{j : (i,j) \in \tilde{\mathcal{E}},\ \forall j \notin \mathcal{S}_v\}$. $\diamondsuit$

Then, for each link $(y,z) \in \tilde{\mathcal{E}}_f$, find a node that is a neighbour of y and z in $\tilde{\mathcal{G}}$ and does not belong to $\mathcal{S}_v$. By defining $\Theta := \{\theta : \theta \in \tilde{\mathcal{M}}_y \cap \tilde{\mathcal{M}}_z,\ \forall (y,z) \in \tilde{\mathcal{E}}_f\}$ and $\tilde{\mathcal{E}}_\theta := \{(\theta, y), (\theta, z),\ \forall \theta \in \Theta,\ \forall (y,z) \in \tilde{\mathcal{E}}_f\}$, the updates of the sub-graph $\tilde{\mathcal{G}}_v$ are $\tilde{\mathcal{S}}_v \leftarrow \tilde{\mathcal{S}}_v \cap \Theta$ and $\tilde{\mathcal{E}}_v \leftarrow \{\tilde{\mathcal{E}}_v \backslash \tilde{\mathcal{E}}_f\} \cap \tilde{\mathcal{E}}_\theta$.

Since the sub-graph $\tilde{\mathcal{G}}_v$ is modified, Assumption 3 no longer holds. If the number of links that fail, $|\tilde{\mathcal{E}}_f|$, is known by the source node, v, then it can easily update $|\tilde{\mathcal{S}}_v|$ after the reconfiguration since one node is added for each link that fails. If this is not the case, the standard distributed consensus algorithm can be used to recalculate the total number of nodes in $|\tilde{\mathcal{S}}_v|$. To this end, consider an auxiliary variable, denoted by $\xi_{i,t}$, for each node $i \in \mathcal{S}_v$. Initialize $\xi_{i,0} = 0$ for $i \neq v$ and $\xi_{v,0} = 1$. By applying the standard distributed consensus [28]

$$\dot{\xi}_{i,t} = \sum_{j \in \tilde{\mathcal{N}}_i} \alpha(\xi_{j,t} - \xi_{i,t}), \tag{12}$$

$\xi_{i,t}$ converges to $\xi^\star = 1/|\mathcal{S}_v|$ as $t \to \infty$, for all $i \in \mathcal{S}_v$. Therefore, in a finite time, t_s, which denotes the settling time of the consensus, the v^{th} node can obtain $|\mathcal{S}_v| = \text{rnd}(1/\xi_{v,t_s})$. Note that the whole reconfiguration procedure can be performed in a distributed fashion, i.e., the algorithm only requires each node to have local information of its neighbours (Assumption 4) and neighbour-to-neighbour communication. The distributed reconfiguration procedure is stated in Algorithm 1.

Algorithm 1 Reconfiguration Procedure

1: Each node $i \in \mathcal{S}_v$ detects whether there are link failures.
2: **for** nodes y and z, for all $(y,z) \in \tilde{\mathcal{E}}_f$ **do**
3: Detect the links that are failed.
4: Send information of the failed links to the neighbors that do not belong to $\mathcal{S}_v$, e.g., node y sends to $\{j : j \in \tilde{\mathcal{M}}_y\}$ and node z sends to $\{j : j \in \tilde{\mathcal{M}}_z\}$.
5: **end for**
6: **for** $\theta \in \Theta$ **do**
7: Receive the same information of one failed link from two different neighbors.
8: Confirm to join the sub-graph $\tilde{\mathcal{G}}_v$.
9: **end for**
10: Compute the updated $|\mathcal{S}_v|$ using the distributed consensus algorithm.

Remark 5: The intersection of $\tilde{\mathcal{M}}_y$ and $\tilde{\mathcal{M}}_z$ for $(y,z) \in \tilde{\mathcal{E}}_f$ might be empty. In this case, the associated information-sharing graph cannot be reconfigured. $\diamondsuit$

IV. Application of the Proposed Protocol to DMPC in Microgrids

In this section, the optimal power allocation problem of a network of microgrids is discussed and solved by applying a distributed MPC strategy. It is a high-level control problem with the objective to determine the set-points of the distributed generators. Furthermore, it is assumed that there are low-level local controllers that ensure the operation of distributed generators is according to the obtained set points [29]. Some previous works, e.g., [21], [22], and [30], have addressed the advantages of using distributed control strategy for energy management in power networks. Since such systems cover a large geographical area, the communication between local controllers is important to be maintained. Therefore, it is thought that the proposed information-exchange protocol is suitable to be applied in these systems.

A. System Description

Consider the system as an undirected graph $\mathcal{G} = (\mathcal{S}, \mathcal{E})$, where $\mathcal{S} = \{1, \ldots, n\}$ denotes the set of microgrids, and $\mathcal{E}$ denotes the physical interconnection among the microgrids, i.e., the link $(i,j) \in \mathcal{E}$ corresponds to the possibility of exchanging energy between the i^{th} and the j^{th} microgrids. Note that $\mathcal{N}_i$ denotes the set of the neighbours of the i^{th} microgrid. Each microgrid is considered to have local loads, a dispatchable distributed generator, a storage system, and a controller that must provides set points to the generator. Furthermore, it is possible to transfer energy between two connected microgrids and there exists an external party from which the microgrids can import energy if necessary. Note that the consideration of having one generator at each microgrid is to simplify the problem without loss of generality. The following formulation can easily be extended for the case in which there are multiple distributed generators per microgrid.

Each microgrid, $i \in \mathcal{S}$, has a power balance equation that must be satisfied at each time instant, as follows:

$$p^{\text{d}}_{i,k} - p^{\text{g}}_{i,k} - p^{\text{st}}_{i,k} - p^{\text{im}}_{i,k} - \sum_{j \in \mathcal{N}_i} p^{\text{t}}_{ij,k} = 0, \tag{13}$$

where $p^{\text{d}}_{i,k} \in \mathbb{R}_{\geq 0}$ denotes the local demand (load) of the i^{th} microgrid, $p^{\text{g}}_{i,k} \in \mathbb{R}_{\geq 0}$ denotes the total generated power of the dispatchable generator, $p^{\text{st}}_{i,k} \in \mathbb{R}$ denotes the power delivered by or to the storage, $p^{\text{im}}_{i,k} \in \mathbb{R}_{\geq 0}$ denotes the imported power from an external party, and $p^{\text{t}}_{ij,k} \in \mathbb{R}$ is the power that is transferred between the i^{th} microgrid and its neighbour $j \in \mathcal{N}_i$. The storage systems have dynamical behaviour that is described by a discrete-time state-space model, i.e.,

$$x_{i,k+1} = a_i x_{i,k} + b_i p^{\text{st}}_{i,k}, \quad \forall i \in \mathcal{S}, \tag{14}$$

where x_i is the state-of-charge (SOC), a_i denotes the efficiency of the storage, $0 < a_i < 1$, and $b_i = -T_{\text{s}}/e^{\max}_i$, where $T_{\text{s}} \in \mathbb{R}_{\geq 0}$ is the sampling time and $e^{\max}_i$ denotes the maximum energy that can be stored. The operational

constraints that are imposed to the storages are the upper and lower bounds of the SOC, as follows:

$$x_i^{\min} \leq x_{i,k} \leq x_i^{\max}, \quad -p_i^{\mathrm{ch}} \leq p_{i,k}^{\mathrm{st}} \leq p_i^{\mathrm{dc}}, \quad \forall i \in \mathcal{S}, \ (15)$$

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

Furthermore, some operational constraints of the power generated by the distributed generators $p_{i,k}^{\mathrm{g}} \in \mathbb{R}$, the power transferred between two neighbouring microgrids, and the power imported from the external party are also established, as follows:

$$p_i^{\mathrm{g,min}} \leq p_{i,k}^{\mathrm{g}} \leq p_i^{\mathrm{g,max}}, \quad \forall i \in \mathcal{S}, \quad (16)$$

$$-p_i^{\mathrm{g,down}} \leq p_{i,k}^{\mathrm{g}} - p_{i,k-1}^{\mathrm{g}} \leq p_i^{\mathrm{g,up}}, \quad \forall i \in \mathcal{S}, \quad (17)$$

$$-p_{ij}^{\mathrm{t,max}} \leq p_{ij,k}^{\mathrm{t}} \leq p_{ij}^{\mathrm{t,max}}, \quad \forall (i,j) \in \mathcal{E}, \quad (18)$$

$$p_{i,k}^{\mathrm{im}} \leq p^{\mathrm{im,max}}, \quad \forall i \in \mathcal{S}, \quad (19)$$

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, $p_i^{\mathrm{g,down}}, p_i^{\mathrm{g,up}} \in \mathbb{R}_{\geq 0}$ denote the maximum ramping down and ramping up of the power, respectively, while $p^{\mathrm{t,max}}, p^{\mathrm{im,max}} \in \mathbb{R}_{\geq 0}$ is the upper bound of the transferred power and imported power from the external party. In addition, there are additional coupled constraints related to $p_{ij,k}^{\mathrm{t}}$, which are

$$p_{ij,k}^{\mathrm{t}} + p_{ji,k}^{\mathrm{t}} = 0, \quad i \neq j, \ \forall (i,j) \in \mathcal{E}. \quad (20)$$

Let the vector $\boldsymbol{u}_{i,k} = [\boldsymbol{u}_{i,k}^{l\top} \ \boldsymbol{u}_{i,k}^{c\top}]^\top \in \mathbb{R}^{n_{u,i}}$ with $\boldsymbol{u}_{i,k}^{l} = [p_{i,k}^{\mathrm{st}} \ p_{i,k}^{\mathrm{g}} \ p_{i,k}^{\mathrm{im}}]^\top$and $\boldsymbol{u}_{i,k}^{c} = [p_{ij,k}^{\mathrm{t}}]_{j \in \mathcal{N}_i}$ be the decision/control inputs. Denote the trajectory of the control inputs and that of the coupled control inputs by $\tilde{\boldsymbol{u}}_{i,k}$ and $\tilde{\boldsymbol{u}}_{i,k}^{c}$, respectively. Moreover, consider a quadratic cost, i.e., $J_{i,k}(\tilde{\boldsymbol{u}}_{i,k}) = \sum_{\ell=k}^{k+h_p-1} \boldsymbol{u}_{i,\ell|k}^\top R_i \boldsymbol{u}_{i,\ell|k}$, where $R_i = \mathrm{diag}\left([c_i^{\mathrm{st}} \ c_i^{\mathrm{g}} \ c_i^{\mathrm{im}} \ c_i^{\mathrm{t}} \mathbb{1}_{|\mathcal{N}_i|}^\top]^\top\right) \succ 0$, for all $i \in \mathcal{S}$. Note that c_i^{st}, c_i^{g}, c_i^{im}, and c_i^{t} denote the cost to store, generate, import, and transfer energy, respectively. The centralized optimization problem that is convex and needs to be solved in an MPC scheme can be written as

$$\underset{\{\tilde{\boldsymbol{u}}_{i,k}\}_{i\in\mathcal{S}}}{\text{minimize}} \quad \sum_{i\in\mathcal{S}} J_{i,\ell}(\tilde{\boldsymbol{u}}_{i,k}) \quad (21a)$$

$$\text{subject to} \quad \tilde{\boldsymbol{x}}_{i,k+1} = \boldsymbol{a}_i x_{i,k} + \boldsymbol{B}_i \tilde{\boldsymbol{u}}_{i,k}, \ \forall i \in \mathcal{S}, \quad (21b)$$

$$\boldsymbol{H}_i \tilde{\boldsymbol{u}}_{i,k} \leq \boldsymbol{h}_{i,k}, \ \forall i \in \mathcal{S}, \quad (21c)$$

$$\tilde{\boldsymbol{u}}_{i,k}^{c} + \sum_{j\in\mathcal{N}_i} \boldsymbol{G}_{ij} \tilde{\boldsymbol{u}}_{j,k}^{c} = \boldsymbol{0}, \ \forall i \in \mathcal{S}, \quad (21d)$$

where $\tilde{\boldsymbol{x}}_{i,k+1} = \left[\boldsymbol{x}_{i,k+1|k}^\top \ \boldsymbol{x}_{i,k+2|k}^\top \ \cdots \ \boldsymbol{x}_{i,k+h_p|k}^\top\right]^\top$ is the trajectory of the state over the whole prediction horizon. The dynamics of the SOC of the storages (14) are rewritten as $x_{i,k+1} = a_i x_{i,k} + \boldsymbol{b}_i \boldsymbol{u}_{i,k}$, in which $\boldsymbol{b}_i = [b_i \ \boldsymbol{0}_{2+|\mathcal{N}_i|}]$ in order to form (21b) with the appropriate concatenated $\boldsymbol{a}_i \in \mathbb{R}^{h_p \times 1}$ and $\boldsymbol{B}_i \in \mathbb{R}^{h_p \times h_p n_{u,i}}$.

The constraints that only include local decisions, which are (13), (15)–(19), over the whole prediction horizon, are compacted in (21c) with the appropriate $\boldsymbol{H}_i$ and $\boldsymbol{h}_{i,k}$. Moreover, the coupled constraints in (20) for the whole prediction horizon are written as in (21d). Additionally, it is assumed that Problem (21) has a set of feasible solutions for $k \in \mathbb{Z}_{\geq 0}$.

B. DMPC Design

The DMPC strategy that is based on dual decomposition is designed for Problem (21). It is presented in Algorithm 2, where $\boldsymbol{\lambda}_i \in \mathbb{R}^{h_p|\mathcal{N}_i|}$, for all $i \in \mathcal{S}$, are the Lagrange multipliers associated to the coupled constraints (21d) and $\boldsymbol{\psi}_i = \left(\tilde{\boldsymbol{u}}_{i,k}^{c} + \sum_{j\in\mathcal{N}_i} \boldsymbol{G}_{ij} \tilde{\boldsymbol{u}}_{j,k}^{c}\right)$, for all $i \in \mathcal{S}$. Notice that Algorithm 2 requires the microgrids to exchange information with their neighbours twice at each iteration, as stated in steps 3 and 5. It is assumed that the default protocol of exchanging information is that a microgrid sends and receives information to and from the neighbours through the direct communication links available between them. Furthermore, the proposed information-exchange protocol that is based on DPD may also be applied to this algorithm to replace the default one. When the DPD-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. In addition, $r_{\max}$, which denotes the maximum number of iterations available in one sampling time, depends on the computational power of the controllers.

Algorithm 2 The Dual-Ascent Algorithm

1: All sub-systems, $i \in \mathcal{S}$, set $r = 1$ and initialize $\boldsymbol{\lambda}_i^{(r)}$.
2: **while** $r < r_{\max}$ **do**
3: *Information exchange:* Receive $\boldsymbol{\lambda}_j^{(r)}$ from the neighbours, all $j \in \mathcal{N}_i$, and send $\boldsymbol{\lambda}_i^{(r)}$ to the neighbours.
4: Solve the local optimization problem:

$$\underset{\tilde{\boldsymbol{u}}_{i,k}}{\text{minimize}} \quad f_i(\tilde{\boldsymbol{u}}_{i,k}) + \tilde{\boldsymbol{y}}_i^\top \tilde{\boldsymbol{u}}_{i,k}^{c}$$
$$\text{subject to (21b) and (21c)},$$

where $\tilde{\boldsymbol{y}}_i^\top = \boldsymbol{\lambda}_i^{(r)\top} + \sum_{j\in\mathcal{N}_i} \boldsymbol{\lambda}_j^{(r)\top} \boldsymbol{G}_{ji}$.
5: *Information exchange:* Receive the solution $\tilde{\boldsymbol{u}}_{j,k}^{c}$ from the neighbours, all $j \in \mathcal{N}_i$, and send $\tilde{\boldsymbol{u}}_{i,k}^{c}$ to the neighbours.
6: Update $\boldsymbol{\lambda}_i$ as $\boldsymbol{\lambda}_i^{(r+1)} = \boldsymbol{\lambda}_i^{(r)} + \gamma \boldsymbol{\psi}_i$, for $0 < \gamma < 1$.
7: $r \leftarrow r + 1$.
8: **end while**

C. Simulation Results and Discussion

Numerical simulations are carried out in MATLAB on a PC with 16 GB of RAM and 2.6 GHz Intel core i7. The power network consists of 12 microgrids, i.e., $\mathcal{S} = \{1, \ldots, 12\}$. Fig. 1 depicts the topology of the grid, denoted by $\mathcal{G} = (\mathcal{S}, \mathcal{E})$, and the overall information-sharing network of the controllers, denoted by $\tilde{\mathcal{G}} = (\mathcal{S}, \tilde{\mathcal{E}})$. The simulation time of all simulations is one day with the sampling time of 15 minutes. The prediction horizon of the DMPC controller

TABLE I

PARAMETERS OF THE MICROGRIDS

Parameters	Value	Unit	Sub-system (i)
$x_i^{\min}$, $x_i^{\max}$	20%, 80%	-	all
p_i^{ch}, p_i^{dc}	50, 50	kW	all
$p_i^{\mathrm{g,min}}$, $p_i^{\mathrm{g,max}}$	0, 1000	kW	all
$p_i^{\mathrm{g,down}}$, $p_i^{\mathrm{g,up}}$	100, 100	kW	all
$p_i^{\mathrm{t,max}}$, $p_i^{\mathrm{im,max}}$,	100, 2000	kW	all
a_i	0.90	-	all
c_i^{st}, c_i^{im}, c_i^{t}	0.1, 250, 0.1	-	all
c_i^{g}	2	-	2, 6, 11
c_i^{g}	10	-	1, 3, 4, 5, 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	DPD-based, no failures	1.00
3	Default, with failures	1.16
4	DPD-based, with failures	1.00

is $h_p = 6$ and the parameters corresponding to each microgrid are shown in Table I. Furthermore, it is assumed that each local controller knows the local load and its forecast over the prediction horizon at each time instant.

In order to compare the performance of the proposed method, four scenarios, which are described in Table II, are simulated. The communication failures are defined as follows. At $k = \{2, 3, 10-13, 40-42, 80-82\}$, the information-sharing network during the failure scenarios is $\tilde{\mathcal{G}}' = (\mathcal{S}, \tilde{\mathcal{E}}')$, where $\tilde{\mathcal{E}}' = \tilde{\mathcal{E}} \backslash \{(2,3),(4,6),(9,11)\}$ (the failed links are indicated by $||$ in Fig. 1). In addition, in Scenarios 2 and 4, the proposed information-exchange protocol uses sub-graphs as shown in Fig. 2 as the information-sharing graphs in order to increase the convergence rates. During failures, all information-sharing sub-graphs only lose one link, hence they are still connected.

Scenario 1 is considered as the baseline performance since the controllers produce the global optimal solution. The simulation result of Scenario 2, as seen in Table II, shows that the proposed information-exchange protocol is able to achieve the optimal performance. In Scenario 3, the microgrids that are disconnected cannot exchange information using the default protocol. Therefore, these microgrids adopt Assumption 1, which means that the unknown information of the neighbours is considered to be null. Table II shows that there is a performance degradation (16% of higher cost than the optimal case) when the microgrids are in Scenario 3. On the other hand, all microgrids are able to obtain the required information with the proposed methodology despite the occurrence of the failures, as expected. Therefore, the optimal solutions can be obtained by the system, as can be seen by comparing the cost of Scenario 4 and that of Scenario 1, which are equal.

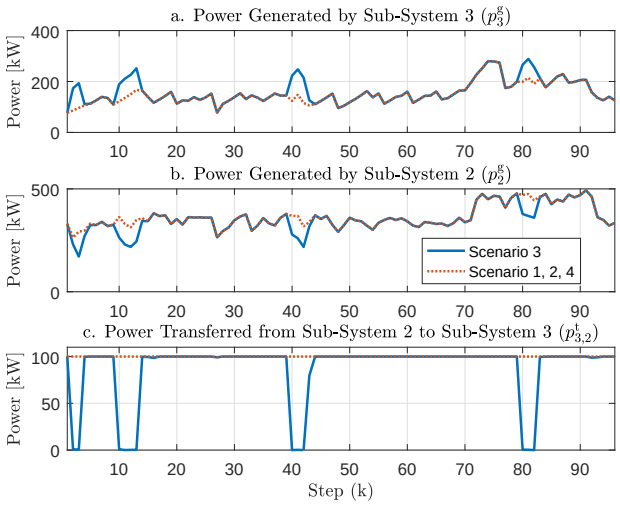

Fig. 3. The operation of sub-systems 3 and 2 in Scenario 3 (solid lines) and Scenarios 1, 2, and 4 (dashed lines): a. Power generated by sub-system 3 (p_3^{g}), b. Power generated by sub-system 2 (p_2^{g}), and c. Power transferred from sub-system 2 to sub-system 3 ($p_{3,2}^{\mathrm{t}}$).

As illustrations, Figs. 3-5 show the operation of the microgrids that are affected by the communication failures in Scenarios 3 and 4. The communication links between these neighbouring microgrids are broken. In Scenario 4, the 3rd and 4th microgrids import power from the the 2nd and 6th microgrids, respectively, at the time instants when the failure occurs. Furthermore, there is also some energy that is exchanged between the 9th and 11th microgrids during these time instants. These decisions are equal to those that are taken in Scenarios 1 and 2. However, in Scenario 3, the 3rd and 4th microgrids do not import any power from their neighbours (see Figures 3c and 4c). Instead, they produce more power to comply with the load (see Figures 3a and 4a). Moreover, the 9th and 11th microgrids do not always exchange energy and their power generation decisions are slightly different than those in Scenario 4 during these time instants (Figure 5c).

V. CONCLUSION AND FUTURE WORK

A methodology to cope with the problem of communication failures in distributed MPC strategies has been proposed. It involves an information-exchange protocol that is based on DPD. A numerical study, which shows the application of this protocol to a distributed MPC strategy that is designed to optimize the power production in a microgrid network, demonstrates the advantages of the proposed protocol.

As future work, in order to extend the results obtained in Section III-D, a study on how to design an optimal information-sharing network that is able to deal with failures in general while ensuring fast convergence rate of the protocol should be carried out. Moreover, an implementation of time-varying partitioning of the information-sharing net-

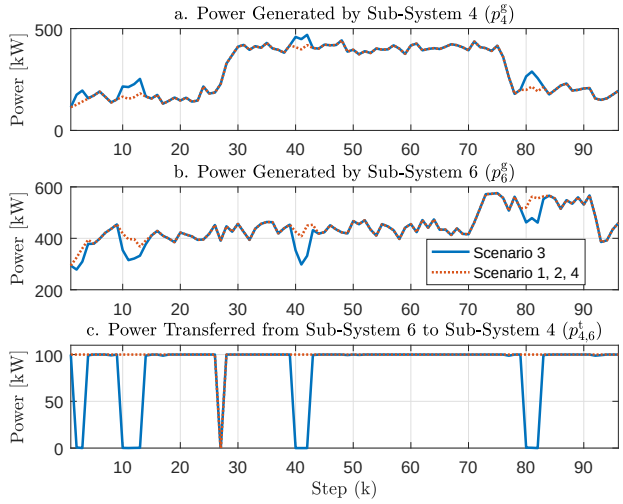

Fig. 4. The operation of sub-systems 4 and 6 in Scenario 3 (solid lines) and Scenarios 1, 2, and 4 (dashed lines): a. Power generated by sub-system 4 (p_4^g), b. Power generated by sub-system 6 (p_6^g), and c. Power transferred from sub-system 6 to sub-system 4 ($p_{4,6}^t$).

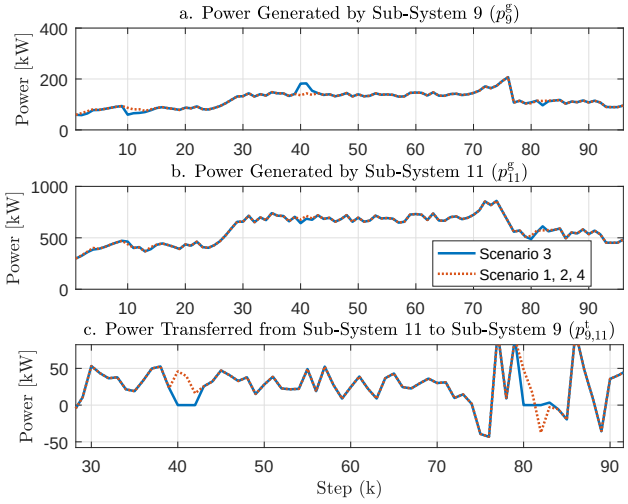

Fig. 5. The operation of sub-systems 9 and 11 in Scenario 3 (solid lines) and Scenarios 1, 2, and 4 (dashed lines): a. Power generated by sub-system 9 (p_9^g), b. Power generated by sub-system 11 (p_{11}^g), and c. Power transferred from sub-system 11 to sub-system 9 ($p_{9,11}^t$).

work may bring advantages to improve the reconfiguration procedure that is discussed in Section III-E.

REFERENCES

[1] C. E. García, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice–A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[2] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

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

[4] 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.

[5] 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.

[6] 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.

[7] X. Liu, Y. Zhang, and K. Y. Lee, "Coordinated distributed MPC for load frequency control of power system with wind farms," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 5140–5150, 2017.

[8] 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.

[9] L. Tarisciotti, G. L. Calzo, A. Gaeta, P. Zanchetta, F. Valencia, and D. Sáez, "A distributed model predictive control strategy for back-to-back converters," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 9, pp. 5867–5878, 2016.

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

[11] S. Li, Y. Zhang, and Q. Zhu, "Nash-optimization enhanced distributed model predictive control applied to the shell benchmark problem," *Information Sciences*, vol. 170, no. 2, pp. 329–349, 2005.

[12] 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.

[13] 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.

[14] M. A. Müller, M. Reble, and F. Allgöwer, "Cooperative control of dynamically decoupled systems via distributed model predictive control," *International Journal of Robust and Nonlinear Control*, vol. 22, no. 12, pp. 1376–1397, 2012.

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

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

[17] J. Barreiro-Gomez, G. Obando, and N. Quijano, "Distributed population dynamics: Optimization and control applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 2, pp. 304–314, 2017.

[18] N. Quijano, C. Ocampo-Martinez, J. Barreiro-Gomez, G. Obando, A. Pantoja, and E. Mojica-Nava, "The role of population games and evolutionary dynamics in distributed control systems: The advantages of evolutionary game theory," *IEEE Control Systems*, vol. 37, no. 1, pp. 70–97, 2017.

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

[20] 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.

[21] M. Kraning, E. Chu, J. Lavaei, and S. Boyd, "Dynamic network energy management via proximal message passing," *Foundations and Trends in Optimization*, vol. 1, no. 2, pp. 73–126, 2014.

[22] 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.

[23] K. Baker, J. Guo, G. Hug, and X. Li, "Distributed MPC for efficient coordination of storage and renewable energy sources across control areas," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 992–1001, 2016.

[24] P. Giselsson, M. D. Doan, T. Keviczky, B. De Schutter, and A. Rantzer, "Accelerated gradient methods and dual decomposition in distributed model predictive control," *Automatica*, vol. 49, no. 3, pp. 829–833, 2013.

[25] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge university press, 2010, vol. 25, no. 3.

[26] F. Chen, Y. Cao, and W. Ren, "Distributed average tracking of multiple time-varying reference signals with bounded derivatives,"

IEEE Transactions on Automatic Control, vol. 57, no. 12, pp. 3169–3174, 2012.

[27] H. K. Khalil, *Nonlinear Control*. Pearson Education, 2015.

[28] 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.

[29] N. Hatziargyriou, Ed., *Microgrids: Architectures and Control*, 1st ed. Wiley/IEEE Press, 2014.

[30] M. Liu, Y. Shi, and X. Liu, "Distributed MPC of aggregated heterogeneous thermostatically controlled loads in smart grid," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1120–1129, 2016.