

A duality-based approach for distributed min-max optimization

Ivano Notarnicola¹, Mauro Franceschelli², Giuseppe Notarstefano¹

Abstract

In this paper we consider a distributed optimization scenario in which a set of processors aims at cooperatively solving a class of min-max optimization problems. This set-up is motivated by peak-demand minimization problems in smart grids. Here, the goal is to minimize the peak value over a finite horizon with: (i) the demand at each time instant being the sum of contributions from different devices, and (ii) the device states at different time instants being coupled through local constraints (e.g., the dynamics). The min-max structure and the double coupling (through the devices and over the time horizon) makes this problem challenging in a distributed set-up (e.g., existing distributed dual decomposition approaches cannot be applied). We propose a distributed algorithm based on the combination of duality methods and properties from min-max optimization. Specifically, we repeatedly apply duality theory and properly introduce ad-hoc slack variables in order to derive a series of equivalent problems. On the resulting problem we apply a dual subgradient method, which turns out to be a distributed algorithm consisting of a minimization on the original primal variables and a suitable dual update. We prove the convergence of the proposed algorithm in objective value. Moreover, we show that every limit point of the primal sequence is an optimal (feasible) solution. Finally, we provide numerical computations for a peak-demand optimization problem in a network of thermostatically controlled loads.

A preliminary short version of this paper is going to appear as [1]. The research leading to these results has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 638992 - OPT4SMART) and from the Italian grant SIR "Scientific Independence of young Researchers", project CoNetDomeSys, code RBSI14OF6H, funded by the Italian Ministry of Research and Education (MIUR).

¹Ivano Notarnicola and Giuseppe Notarstefano are with the Department of Engineering, Università del Salento, Via Monteroni, 73100 Lecce, Italy, `name.lastname@unisalento.it`.

²Mauro Franceschelli is with the Department of Electrical and Electronic Engineering, University of Cagliari, Piazza D'Armi, 09123 Cagliari, Italy, `mauro.franceschelli@diee.unica.it`.

I. INTRODUCTION

Distributed optimization problems arise as building blocks of several network problems in different areas as, e.g., control, estimation and learning. On this regard, the addition of processing, measurement, communication and control capability to the electric power grid is leading to “smart grids” in which tasks, that were typically performed at a central level, can be more efficiently performed by smart devices in a cooperative way. Therefore, these complex systems represent a rich source of motivating optimization scenarios. An interesting example is the design of smart generators, accumulators and loads that cooperatively execute Demand Side Management (DSM) programs [2]. The goal is to reduce the hourly and daily variations and peaks of electric demand by optimizing generation, storage and consumption. A widely adopted objective in DSM programs is Peak-to-Average Ratio (PAR), defined as the ratio between peak-daily and average-daily power demands. PAR minimization gives rise to a min-max optimization problem if the average daily electric load is assumed not to be affected by the demand response strategy.

This problem has been already investigated in the literature in a noncooperative framework. In [3] the authors propose a game-theoretic model for PAR minimization and provide a distributed energy-cost-based strategy for the users which is proven to be optimal. A noncooperative-game approach is also proposed in [4], where optimal strategies are characterized and a distributed scheme is designed based on a proximal decomposition algorithm. It is worth pointing out that in the literature above the term “distributed” is used to indicate that data are deployed on a set of devices, which perform local computation simultaneously. However, the nodes do not run a “distributed algorithm”, that is they do not cooperate and do not exchange information locally over a communication graph.

Motivated by this application scenario, in this paper we propose a novel distributed optimization framework for min-max optimization problems commonly found in DSM problems. Differently from the references above, we consider a cooperative, distributed computation model in which the agents in the network solve the optimization problem (i) without any knowledge of aggregate quantities, (ii) by communicating only with neighboring agents, and (iii) by performing local computations (with no central coordinator).

The distributed algorithm proposed in the paper heavily relies on duality theory. Duality is a widely used tool for parallel and (classical) distributed optimization algorithms as shown, e.g., in the tutorials [5], [6]. More recently, in [7] a distributed, consensus-based, primal-dual algorithm

is proposed to solve constrained optimization problems with separable convex costs and common convex constraints. In [8] the authors use the same technique to solve optimization problems with coupled smooth convex costs and convex inequality constraints. In the proposed algorithm, agents employ a consensus technique to estimate the global cost and constraint functions and use a local primal-dual perturbed subgradient method to obtain a global optimum. These approaches do not apply to optimization problems as the one considered in this paper.

Primal recovery is a key issue in dual methods, since the primal sequence is not guaranteed, in general, to satisfy the dualized primal constraint. Thus, several strategies have been proposed to cope with this issue. In [9], the authors propose and analyze a centralized algorithm for generating approximate primal solutions via a dual subgradient method applied to a convex constrained optimization problem. Moreover, in the paper the problem of (exact) primal recovery and rate analysis of existing techniques is widely discussed. In [10], still in a centralized set-up, the primal convergence rate of dual first-order methods is studied when the primal problem is only approximately solved. In [11] a distributed algorithm is proposed to generate approximate dual solutions for a problem with separable cost function and coupling constraints. A similar optimization set-up is considered in [12] in a distributed set-up. A dual decomposition approach combined with a proximal minimization is proposed to generate a dual solution. In the last two papers, a primal recovery mechanism is proposed to obtain a primal optimal solution.

Another tool used to develop and analyze the distributed algorithm we propose in the paper is min-max optimization, which is strictly related to saddle-point problems. In [13] the authors propose a subgradient method to generate approximate saddle-points. A min-max problem is also considered in [14] and a distributed algorithm based on a suitable penalty approach has been proposed. Differently from our set-up, in [14] each term of the max-function is local and entirely known by a single agent. Another class of algorithms exploits the exchange of active constraints among the network nodes to solve constrained optimization problems which include min-max problems, [15], [16]. Although they work under asynchronous, directed communication they do not scale in set-ups as the one in this paper, in which the terms of the max function are coupled. Very recently, in [17] the authors proposed a distributed projected subgradient method to solve constrained saddle-point problems with agreement constraints. The proposed algorithm is based on saddle-point dynamics with Laplacian averaging. Although our problem set-up fits in those considered in [17], our algorithmic approach and the analysis are different. In [18], [19] saddle point dynamics are used to design distributed algorithms for standard separable optimization

problems.

The main contributions of this paper are as follows. First, we propose a novel distributed optimization framework which is strongly motivated by peak power-demand minimization in DSM. The optimization problem has a min-max structure with local constraints at each node. Each term in the max function represents a daily cost (so that the maximum over a given horizon needs to be minimized), while the local constraints are due to the local dynamics and state/input constraints of the subsystems in the smart grid. The problem is challenging when approached in a distributed way since it is *doubly coupled*. Each term of the max function is coupled among the agents, since it is the sum of local functions each one known by the local agent only. Moreover, the local constraints impose a coupling between different “days” in the time-horizon. The goal is to solve the problem in a distributed computation framework, in which each agent only knows its local constraint and its local objective function at each day.

Second, as main paper contribution, we propose a distributed algorithm to solve this class of min-max optimization problems. The algorithm has a very simple and clean structure in which a primal minimization and a dual update are performed. The primal problem has a similar structure to the centralized one. Despite this simple structure, which resembles standard distributed dual methods, the algorithm is not a standard decomposition scheme [6], and the derivation of the algorithm is non-obvious. Specifically, the algorithm is derived by heavily resorting to duality theory and properties of min-max optimization (or saddle-point) problems. In particular, a sequence of equivalent problems is derived in order to decompose the originally coupled problem into locally-coupled subproblems, and thus being able to design a distributed algorithm. An interesting feature of the algorithm is its expression in terms of the original primal variables and of dual variables arising from two different (dual) problems. Since we apply duality more than once, and on different problems, this property, although apparently intuitive, was not obvious a priori. Another appealing feature of the algorithm is that every limit point of the primal sequence at each node is a (feasible) optimal solution of the original optimization problem (although this is only convex and not strictly convex). This property is obtained by the minimizing sequence of the local primal subproblems without resorting to averaging schemes. Finally, since each node only computes the decision variable of interest, our algorithm can solve both large-scale (many agents are present) and big-data (a large horizon is considered) problems.

The paper is structured as follows. In Section II we formalize our distributed min-max optimization set-up and present the main contribution of the paper, a novel, duality-based

distributed optimization method. In Section III we characterize its convergence properties. In Section IV we corroborate the theoretical results with a numerical example involving peak power minimization in a smart-grid scenario. Finally, in Appendix we provide some useful preliminaries from optimization, specifically basics on duality theory and a result for the subgradient method.

II. PROBLEM SET-UP AND DISTRIBUTED OPTIMIZATION ALGORITHM

In this section we set-up the distributed min-max optimization framework and propose a novel distributed algorithm to solve it.

A. Distributed min-max optimization set-up

We consider a network of N processors which communicate according to a *connected*, undirected graph $\mathcal{G} = (\{1, \dots, N\}, \mathcal{E})$, where $\mathcal{E} \subseteq \{1, \dots, N\} \times \{1, \dots, N\}$ is the set of edges. That is, the edge (i, j) models the fact that node i and j exchange information. We denote by \mathcal{N}_i the set of *neighbors* of node i in the fixed graph \mathcal{G} , i.e., $\mathcal{N}_i := \{j \in \{1, \dots, N\} \mid (i, j) \in \mathcal{E}\}$. Also, we denote by a_{ij} the element i, j of the adjacency matrix. We recall that $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and $i \neq j$, and $a_{ij} = 0$ otherwise.

Next, we introduce the min-max optimization problem to be solved by the network processors in a distributed way. Specifically, we associate to each processor i a decision vector $\mathbf{x}^i = [\mathbf{x}_1^i, \dots, \mathbf{x}_S^i]^\top \in \mathbb{R}^S$, a constraint set $X^i \subseteq \mathbb{R}^S$ and local functions g_s^i , $s \in \{1, \dots, S\}$, and set-up the following optimization problem

$$\begin{aligned} \min_{\mathbf{x}^1, \dots, \mathbf{x}^N} \max_{s \in \{1, \dots, S\}} \sum_{i=1}^N g_s^i(\mathbf{x}_s^i) \\ \text{subj. to } \mathbf{x}^i \in X^i, \quad i \in \{1, \dots, N\} \end{aligned} \tag{1}$$

where for each $i \in \{1, \dots, N\}$ the set $X^i \subseteq \mathbb{R}^S$ is nonempty, convex and compact, and the functions $g_s^i : \mathbb{R} \rightarrow \mathbb{R}$, $s \in \{1, \dots, S\}$, are convex.

Note that we use the superscript $i \in \{1, \dots, N\}$ to indicate that a vector $\mathbf{x}^i \in \mathbb{R}^S$ belongs to node i , while we use the subscript $s \in \{1, \dots, S\}$ to identify a vector component, i.e., \mathbf{x}_s^i is the s -th component of \mathbf{x}^i .

Using a standard approach for min-max problems, we introduce an auxiliary variable P to write the so called epigraph representation of problem (1), given by

$$\begin{aligned} \min_{\mathbf{x}^1, \dots, \mathbf{x}^N, P} \quad & P \\ \text{subj. to } \quad & \mathbf{x}^i \in X^i, \quad i \in \{1, \dots, N\} \\ & \sum_{i=1}^N g_s^i(\mathbf{x}_s^i) \leq P, \quad s \in \{1, \dots, S\}. \end{aligned} \quad (2)$$

It is worth noticing that this problem has a particular structure, which gives rise to interesting challenges in a distributed set-up. First of all, two types of couplings are present, which involve simultaneously the N agents and the S components of each decision variable \mathbf{x}^i . Specifically, for a given index s , the constraint $\sum_{i=1}^N g_s^i(\mathbf{x}_s^i) \leq P$ couples all the vectors \mathbf{x}^i , $i \in \{1, \dots, N\}$. At the same time, for a given $i \in \{1, \dots, N\}$, the constraint X^i couples all the components $\mathbf{x}_1^i, \dots, \mathbf{x}_S^i$ of \mathbf{x}^i . Figure 1 provides a nice graphical representation of this interlaced coupling. Moreover, the problem is both large-scale and big-data. That is, both the number of decision variables and the number of constraints depend on N (and thus scale badly with the number of agents in the network). Also, the dimension of the coupling constraint, S , can be large. Therefore, common approaches as reaching a consensus among the nodes on an optimal solution and/or exchanging constraints are not computationally affordable.

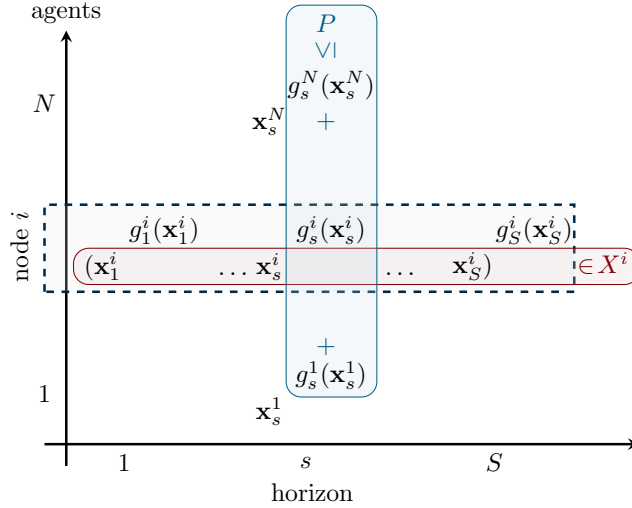


Fig. 1. Graphical representation of interlaced constraints.

To conclude this section, notice that problem (2) is convex, but not strictly convex. This means that it is not guaranteed to have a unique optimal solution. As discussed in the introduction, this

impacts on dual approaches when trying to recover a primal optimal solution, see e.g., [9] and references therein. This aspect is even more delicate in a distributed set-up in which nodes only know part of the constraints and of the objective function.

B. Distributed Duality-Based Peak Minimization (DDPM)

Next, we introduce our distributed optimization algorithm. Informally, the algorithm consists of a two-step procedure. First, each node $i \in \{1, \dots, N\}$ stores a set of variables $((\mathbf{x}^i, \rho^i), \boldsymbol{\mu}^i)$ obtained as a primal-dual optimal solution pair of a local optimization problem with an epigraph structure as the centralized problem. The coupling with the other nodes in the original formulation is replaced by a term depending on neighboring variables $\boldsymbol{\lambda}^{ij}, j \in \mathcal{N}_i$. These variables are updated in the second step according to a suitable linear law weighting the difference of neighboring $\boldsymbol{\mu}^i$. Nodes use a diminishing step-size denoted by $\gamma(t)$ and can initialize the variables $\boldsymbol{\lambda}^{ij}, j \in \mathcal{N}_i$ to arbitrary values. In the next table we formally state our Distributed Duality-Based Peak Minimization (DDPM) algorithm from the perspective of node i .

Distributed Algorithm DDPM

Processor states: $(\mathbf{x}^i, \rho^i), \boldsymbol{\mu}^i$ and $\boldsymbol{\lambda}^{ij}$ for $j \in \mathcal{N}_i$

Evolution:

Gather $\boldsymbol{\lambda}^{ji}(t)$ from $j \in \mathcal{N}_i$

Compute $((\mathbf{x}^i(t+1), \rho^i(t+1)), \boldsymbol{\mu}^i(t+1))$ as a primal-dual optimal solution pair of

$$\begin{aligned} & \min_{\mathbf{x}^i, \rho^i} \rho^i \\ & \text{subj. to } \mathbf{x}^i \in X^i \\ & g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\boldsymbol{\lambda}^{ij}(t) - \boldsymbol{\lambda}^{ji}(t))_s \leq \rho^i, \\ & s \in \{1, \dots, S\} \end{aligned} \tag{3}$$

Gather $\boldsymbol{\mu}^j(t+1)$ from $j \in \mathcal{N}_i$

Update for all $j \in \mathcal{N}_i$

$$\boldsymbol{\lambda}^{ij}(t+1) = \boldsymbol{\lambda}^{ij}(t) - \gamma(t)(\boldsymbol{\mu}^i(t+1) - \boldsymbol{\mu}^j(t+1)) \tag{4}$$

The structure of the algorithm and the meaning of the updates will be clear in the constructive analysis carried out in the next section. At this point we want to point out that although

problem (3) has the same epigraph structure of problem (2), ρ^i is not a copy of the centralized cost P , but rather a local contribution to that cost. That is, as we will see, the total cost P will be the sum of the ρ^i s.

III. ALGORITHM ANALYSIS

The analysis of the proposed DDPM distributed algorithm is constructive and heavily relies on duality theory tools.

We start by deriving the equivalent dual problem of (2) which is formally stated in the next lemma.

Lemma 3.1: The optimization problem

$$\begin{aligned} \max_{\boldsymbol{\mu} \in \mathbb{R}^S} \quad & \sum_{i=1}^N q^i(\boldsymbol{\mu}) \\ \text{subj. to} \quad & \mathbf{1}^\top \boldsymbol{\mu} = 1, \boldsymbol{\mu} \succeq 0, \end{aligned} \quad (5)$$

with $\mathbf{1} := [1, \dots, 1]^\top \in \mathbb{R}^S$ and

$$q^i(\boldsymbol{\mu}) := \min_{\mathbf{x}^i \in X^i} \sum_{s=1}^S \boldsymbol{\mu}_s g_s^i(\mathbf{x}_s^i), \quad (6)$$

for all $i \in \{1, \dots, N\}$, is the dual of problem (2).

Moreover, both problems (2) and (5) have finite optimal cost, respectively P^* and q^* , and strong duality holds, i.e.,

$$P^* = q^*.$$

Proof: We start showing that problem (5) is the dual of (2). Let $\boldsymbol{\mu} := [\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_S]^\top \in \mathbb{R}^S$ be S Lagrange multipliers associated to the inequality constraints $\sum_{i=1}^N g_s^i(\mathbf{x}_s^i) - P \leq 0$ for $s \in \{1, \dots, S\}$ in (2). Then the partial Lagrangian¹ of problem (2) is given by

$$\begin{aligned} \mathcal{L}_1(\mathbf{x}^1, \dots, \mathbf{x}^N, P, \boldsymbol{\mu}) &= P + \sum_{s=1}^S \boldsymbol{\mu}_s \left(\sum_{i=1}^N g_s^i(\mathbf{x}_s^i) - P \right) \\ &= P \left(1 - \sum_{s=1}^S \boldsymbol{\mu}_s \right) + \sum_{i=1}^N \sum_{s=1}^S \boldsymbol{\mu}_s g_s^i(\mathbf{x}_s^i). \end{aligned}$$

¹We have a “partial Lagrangian” since we do not dualize all the constraints. Here local constraints $\mathbf{x}^i \in X^i$, $i \in \{1, \dots, N\}$, are not dualized.

By definition, the dual function is defined as

$$q(\boldsymbol{\mu}) := \min_{\mathbf{x}^1 \in X_1, \dots, \mathbf{x}^N \in X_N, P} \mathcal{L}_1(\mathbf{x}^1, \dots, \mathbf{x}^N, P, \boldsymbol{\mu}),$$

where the presence of constraints $\mathbf{x}^i \in X^i$ for all $i \in \{1, \dots, N\}$ is due to the fact that we have not dualized them.

The minimization of \mathcal{L}_1 with respect to P gives rise to the simplex constraint $\sum_{s=1}^S \boldsymbol{\mu}_s = 1$. The minimization with respect to \mathbf{x}^i splits over $i \in \{1, \dots, N\}$, so that the dual function can be written as the sum of terms q^i given in (6).

To prove strong duality, we show that the strong duality theorem for convex inequality constraints, [20, Proposition 5.3.1], applies. Since the sets X^i , $i \in \{1, \dots, N\}$, are convex (and compact), we need to show that the inequality constraints $\sum_{i=1}^N g_s^i(\mathbf{x}_s^i) - P \leq 0$ for all $s \in \{1, \dots, S\}$ are convex and that there exist $\bar{\mathbf{x}}^1 \in X_1, \dots, \bar{\mathbf{x}}^N \in X_N$ and \bar{P} such that the strict inequality holds. Since each g_s^i and $-P$ are convex functions, then for all s each function

$$\bar{g}_s(\mathbf{x}_s^1, \dots, \mathbf{x}_s^N, P) := \sum_{i=1}^N g_s^i(\mathbf{x}_s^i) - P$$

is convex. Also, since the sets X^i , $i \in \{1, \dots, N\}$ are nonempty, there exist $\bar{\mathbf{x}}^i \in X^i$, $i \in \{1, \dots, N\}$, and a sufficiently large (finite) \bar{P} such that the strict inequalities $\bar{g}(\bar{\mathbf{x}}_s^1, \dots, \bar{\mathbf{x}}_s^N, \bar{P}) < 0$, $s \in \{1, \dots, S\}$ are satisfied and, thus, the Slater's condition holds. Finally, since a feasible point for the convex problem (1) always exists, then the optimal cost P^* is finite and so q^* , thus concluding the proof. \square

In order to make problem (5) amenable for a distributed solution, we can rewrite it in an equivalent form. To this end, we introduce copies of the common optimization variable $\boldsymbol{\mu}$ and coherence constraints having the sparsity of the connected graph \mathcal{G} , thus obtaining

$$\begin{aligned} & \max_{\boldsymbol{\mu}^1, \dots, \boldsymbol{\mu}^N} \sum_{i=1}^N q^i(\boldsymbol{\mu}^i) \\ & \text{subj. to } \mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0, \quad i \in \{1, \dots, N\} \\ & \quad \boldsymbol{\mu}^i = \boldsymbol{\mu}^j, \quad (i, j) \in \mathcal{E}. \end{aligned} \tag{7}$$

Notice that we have also duplicated the simplex constraint, so that it becomes *local* at each node.

To solve this problem, we can use a dual decomposition approach by designing a dual subgradient algorithm. Notice that dual methods can be applied to (7) since the constraints are convex and the cost function concave. Also, as known in the distributed optimization literature,

a dual subgradient algorithm applied to problem (7) would immediately result into a distributed algorithm if functions q^i were available in closed form.

Remark 3.2: In standard convex optimization deriving the dual of a dual problem brings back to a primal formulation. However, we want to stress that in what we will develop in the following, problem (7) is dualized rather than problem (5). In particular, different constraints are dualized, namely the coherence constraints rather than the simplex ones. Therefore, it is not obvious if and how this leads back to a primal formulation. \square

We start deriving the dual subgradient algorithm by dualizing only the coherence constraints. Thus, we write the partial Lagrangian

$$\begin{aligned} \mathcal{L}_2(\boldsymbol{\mu}^1, \dots, \boldsymbol{\mu}^N, \{\boldsymbol{\lambda}^{ij}\}_{(i,j) \in \mathcal{E}}) \\ = \sum_{i=1}^N \left(q^i(\boldsymbol{\mu}^i) + \sum_{j \in \mathcal{N}_i} \boldsymbol{\lambda}^{ij \top} (\boldsymbol{\mu}^i - \boldsymbol{\mu}^j) \right) \end{aligned} \quad (8)$$

where $\boldsymbol{\lambda}^{ij} \in \mathbb{R}^S$ for all $(i, j) \in \mathcal{E}$ are Lagrange multipliers associated to the constraints $\boldsymbol{\mu}^i - \boldsymbol{\mu}^j = 0$.

Since the communication graph \mathcal{G} is undirected and connected, we can exploit the symmetry of the constraints. In fact, for each $(i, j) \in \mathcal{E}$ we also have $(j, i) \in \mathcal{E}$, and, expanding all the terms in (8), for given i and j , we always have both the terms $\boldsymbol{\lambda}^{ij \top} (\boldsymbol{\mu}^i - \boldsymbol{\mu}^j)$ and $\boldsymbol{\lambda}^{ji \top} (\boldsymbol{\mu}^j - \boldsymbol{\mu}^i)$. Thus, after some simple algebraic manipulations, we get

$$\begin{aligned} \mathcal{L}_2(\boldsymbol{\mu}^1, \dots, \boldsymbol{\mu}^N, \{\boldsymbol{\lambda}^{ij}\}_{(i,j) \in \mathcal{E}}) \\ = \sum_{i=1}^N \left(q^i(\boldsymbol{\mu}^i) + \boldsymbol{\mu}^i \top \sum_{j \in \mathcal{N}_i} (\boldsymbol{\lambda}^{ij} - \boldsymbol{\lambda}^{ji}) \right), \end{aligned} \quad (9)$$

which is separable with respect to $\boldsymbol{\mu}^i$.

The dual of problem (7) is thus

$$\min_{\{\boldsymbol{\lambda}^{ij}\}_{(i,j) \in \mathcal{E}}} \eta(\{\boldsymbol{\lambda}^{ij}\}_{(i,j) \in \mathcal{E}}) = \sum_{i=1}^N \eta^i(\{\boldsymbol{\lambda}^{ij}, \boldsymbol{\lambda}^{ji}\}_{j \in \mathcal{N}_i}), \quad (10)$$

with, for all $i \in \{1, \dots, N\}$,

$$\eta^i(\{\boldsymbol{\lambda}^{ij}, \boldsymbol{\lambda}^{ji}\}_{j \in \mathcal{N}_i}) := \max_{\mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0} q^i(\boldsymbol{\mu}^i) + \boldsymbol{\mu}^i \top \sum_{j \in \mathcal{N}_i} (\boldsymbol{\lambda}^{ij} - \boldsymbol{\lambda}^{ji}). \quad (11)$$

In the next lemma we characterize the properties of problem (10).

Lemma 3.3: Problem (10), which is the dual of problem (7), has a bounded optimal cost, call it η^* , and strong duality holds, so that

$$\eta^* = q^* = P^*. \quad (12)$$

Proof: Since problem (5) is a dual problem, its cost function $\sum_{i=1}^N q^i(\boldsymbol{\mu})$ is concave on its domain, which is convex (simplex constraint). Moreover, by Lemma 3.1 its optimal cost q^* is finite. Problem (7) is an equivalent formulation of (5) and, thus, has the same (finite) optimal cost q^* . This allows us to conclude that strong duality holds between problem (7) and its dual (10), so that, $\eta^* = q^*$. The second equality in (12) holds by Lemma 3.1, so that the proof follows. \square

Problem (10) has a particularly appealing structure for distributed computation. In fact, the cost function is separable and each term η^i of the cost function depends only on neighboring variables $\boldsymbol{\lambda}^{ij}$ and $\boldsymbol{\lambda}^{ji}$ with $j \in \mathcal{N}_i$. Thus, a subgradient method applied to this problem turns out to be a distributed algorithm. Since problem (10) is the dual of (7) we recall, [20, Section 6.1], how to compute a subgradient of η with respect to each component, that is,

$$\frac{\tilde{\partial}\eta(\{\boldsymbol{\lambda}^{ij}\}_{(i,j) \in \mathcal{E}})}{\partial \boldsymbol{\lambda}^{ij}} = \boldsymbol{\mu}^{i*} - \boldsymbol{\mu}^{j*}, \quad (13)$$

where $\frac{\tilde{\partial}\eta(\cdot)}{\partial \boldsymbol{\lambda}^{ij}}$ denotes the component associated to the variable $\boldsymbol{\lambda}^{ij}$ of a subgradient of η , and

$$\boldsymbol{\mu}^{k*} \in \underset{\mathbf{1}^\top \boldsymbol{\mu}^k = 1, \boldsymbol{\mu}^k \succeq 0}{\operatorname{argmax}} \left(q_k(\boldsymbol{\mu}^k) + \boldsymbol{\mu}^{k\top} \sum_{h \in \mathcal{N}_k} (\boldsymbol{\lambda}^{kh} - \boldsymbol{\lambda}^{hk}) \right),$$

for $k = i, j$.

The distributed dual subgradient algorithm for problem (7) can be summarized as follows. For each node $i \in \{1, \dots, N\}$:

(S1) receive $\boldsymbol{\lambda}^{ji}(t)$, for each $j \in \mathcal{N}_i$, and compute a subgradient $\boldsymbol{\mu}^i(t+1)$ by solving

$$\max_{\boldsymbol{\mu}^i} q^i(\boldsymbol{\mu}^i) + \boldsymbol{\mu}^{i\top} \sum_{j \in \mathcal{N}_i} (\boldsymbol{\lambda}^{ij}(t) - \boldsymbol{\lambda}^{ji}(t)) \quad (14)$$

$$\text{subj. to } \mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0.$$

(S2) exchange with neighbors the updated $\boldsymbol{\mu}^j(t+1)$, $j \in \mathcal{N}_i$, and update $\boldsymbol{\lambda}^{ij}$, $j \in \mathcal{N}_i$, via

$$\boldsymbol{\lambda}^{ij}(t+1) = \boldsymbol{\lambda}^{ij}(t) - \gamma(t)(\boldsymbol{\mu}^i(t+1) - \boldsymbol{\mu}^j(t+1)).$$

where $\gamma(t)$ denotes a diminishing step-size satisfying Assumption B.3 in Appendix B.

It is worth noting that in (14) the value of $\lambda^{ij}(t)$ and $\lambda^{ji}(t)$, for $j \in \mathcal{N}_i$, is fixed as highlighted by the index t . Moreover, we want to stress, once again, that the algorithm is *not* implementable as it is written, since functions q^i are not available in closed form.

On this regard, we point out that here we slightly abuse notation since in (S1)-(S2) we use $\mu^i(t)$ as in the DDPM algorithm, but without proving its equivalence yet. Since we will prove it in the next, we preferred not to overweight the notation.

Before proving the convergence of the updates (S1)-(S2) we need the following lemma.

Lemma 3.4: For each $i \in \{1, \dots, N\}$, the function $\mu^i \mapsto q^i(\mu^i)$ defined in (6) is concave over $\mu^i \succeq 0$.

Proof: For each $i \in \{1, \dots, N\}$, consider the (feasibility) convex problem

$$\begin{aligned} & \min_{\mathbf{z}^i \in X^i} 0 \\ & \text{subj. to } g_s^i(\mathbf{z}_s^i) \leq 0, \quad s \in \{1, \dots, S\}. \end{aligned}$$

Then, $q^i(\mu^i)$ is the dual function of that problem and, thus, is a concave function on its domain, namely $\mu^i \succeq 0$. \square

We can now prove the convergence in objective value of the dual subgradient.

Lemma 3.5: The dual subgradient updates (S1)-(S2), with step-size $\gamma(t)$ satisfying Assumption B.3, generate sequences $\{\lambda^{ij}(t)\}$, $(i, j) \in \mathcal{E}$, that converge in objective value to the optimal cost η^* of problem (10).

Proof: As already recalled in equation (13), we can build subgradients of η by solving problem in the form (14). Since in (14), the maximization of the concave (Lemma 3.4) function q^i is performed over the nonempty, compact (and convex) probability simplex $\mathbf{1}^\top \mu^i = 1$, $\mu^i \succeq 0$, then the maximum is always attained at a finite value. As a consequence, at each iteration the subgradients of η are bounded quantities. Moreover, the step-size $\gamma(t)$ satisfies Assumption B.3 and, thus, we can invoke Proposition B.4 which guarantees that (S1)-(S2) converges in objective value to the optimal cost η^* of problem (10) so that the proof follows. \square

We can explicitly rephrase update (14) by plugging in the definition of q^i , given in (6), thus obtaining the following max-min optimization problem

$$\max_{\mathbf{1}^\top \mu^i = 1, \mu^i \succeq 0} \left(\min_{\mathbf{x}^i \in X^i} \sum_{s=1}^S \mu_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right) \right). \quad (15)$$

Notice that (15) is a local problem at each node i once $\lambda^{ij}(t)$ and $\lambda^{ji}(t)$ for all $j \in \mathcal{N}_i$ are given. Thus, the dual subgradient algorithm (S1)-(S2) could be implemented in a distributed

way by letting each node i solve problem (15) and exchange $\lambda^{ij}(t)$ and $\lambda^{ji}(t)$ with neighbors $j \in \mathcal{N}_i$. Next we further explore the structure of (15) to prove that DDPM solves the original problem (2).

The next lemma is a first instrumental result.

Lemma 3.6: Consider the optimization problem

$$\max_{\boldsymbol{\mu}^i} \sum_{s=1}^S \mu_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right) \quad (16)$$

$$\text{subj. to } \mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0,$$

with given \mathbf{x}^i , $\lambda^{ij}(t)$ and $\lambda^{ji}(t)$, $j \in \mathcal{N}_i$. Then, the problem

$$\min_{\rho^i} \rho^i$$

$$\text{subj. to } g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \leq \rho^i, \quad (17)$$

$$s \in \{1, \dots, S\}$$

is dual of (16) and strong duality holds.

Proof: First, since \mathbf{x}^i (as well as $\lambda^{ij}(t)$ and $\lambda^{ji}(t)$) is given, problem (16) is a feasible linear program (the simplex constraint is nonempty) and, thus, strong duality holds. Introducing a scalar multiplier ρ^i associated to the constraint $\mathbf{1}^\top \boldsymbol{\mu}^i = 1$, we write the partial Lagrangian of (16)

$$\mathcal{L}_3(\boldsymbol{\mu}^i, \rho^i) = \sum_{s=1}^S \mu_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right) + \rho^i (1 - \mathbf{1}^\top \boldsymbol{\mu}^i)$$

and rearrange it as

$$\mathcal{L}_3(\boldsymbol{\mu}^i, \rho^i) = \sum_{s=1}^S \mu_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s - \rho^i \right) + \rho^i.$$

The dual function $\max_{\boldsymbol{\mu}^i \succeq 0} \mathcal{L}_3(\boldsymbol{\mu}^i, \rho^i)$ is equal to ρ^i with domain given by the inequalities $\rho^i \geq g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s$, $s \in \{1, \dots, S\}$. Thus, the dual problem is obtained by maximizing the dual function over its domain giving (17), so that the proof follows. \square

The next lemma is a second instrumental result.

Lemma 3.7: Max-min optimization problem (15) is the saddle point problem associated to problem (3)

$$\begin{aligned} & \min_{\mathbf{x}^i, \rho^i} \rho^i \\ & \text{subj. to } \mathbf{x}^i \in X^i \\ & g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \leq \rho^i, \\ & s \in \{1, \dots, S\}. \end{aligned}$$

Moreover, a primal-dual optimal solution pair of (3), call it $((\mathbf{x}^i(t+1), \rho^i(t+1)), \boldsymbol{\mu}^i(t+1))$, exists and $(\mathbf{x}^i(t+1), \boldsymbol{\mu}^i(t+1))$ is a solution of (15).

Proof: We give a constructive proof which clarifies how the problem (3) is derived from (15).

Define

$$\phi(\mathbf{x}^i, \boldsymbol{\mu}^i) := \sum_{s=1}^S \boldsymbol{\mu}_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right) \quad (18)$$

and note that (i) $\phi(\cdot, \boldsymbol{\mu}^i)$ is closed and convex for all $\boldsymbol{\mu}^i \succeq 0$ (affine transformation of a convex function with compact domain X^i) and (ii) $\phi(\mathbf{x}^i, \cdot)$ is closed and concave since it is a linear function with compact domain $(\mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0)$, for all $\mathbf{x}^i \in \mathbb{R}^S$. Thus we can invoke Proposition A.2 which allows us to switch max and min operators, and write

$$\begin{aligned} & \max_{\mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0} \left(\min_{\mathbf{x}^i \in X^i} \sum_{s=1}^S \boldsymbol{\mu}_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right) \right) \\ & = \min_{\mathbf{x}^i \in X^i} \left(\max_{\mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0} \sum_{s=1}^S \boldsymbol{\mu}_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right) \right). \end{aligned} \quad (19)$$

Since the inner maximization problem depends nonlinearly on \mathbf{x}^i (which is itself an optimization variable), it cannot be performed without also considering the simultaneous minimization over \mathbf{x}^i . We overcome this issue by substituting the inner maximization problem with its equivalent dual minimization. In fact, by Lemma 3.6 we can rephrase the right hand side of (19) as

$$\min_{\mathbf{x}^i \in X^i} \left(\min_{\substack{\rho^i : g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \leq \rho^i \\ s \in \{1, \dots, S\}}} \rho^i \right). \quad (20)$$

At this point, a *joint* (constrained) minimization with respect to \mathbf{x}^i and ρ^i can be simultaneously performed leading to problem (3).

To prove the second part, namely that a primal-dual optimal solution pair exists and solves problem (15), we first notice that problem (3) is convex. Indeed, the cost function is linear and the constraints are convex (X^i is convex as well as the functions $g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s$ and $-\rho^i$). Then, by using similar arguments as in Lemma 3.1, we can show that the problem satisfies the Slater's constraint qualification and, thus, strong duality holds. Therefore, a primal-dual optimal solution pair $(\mathbf{x}^i(t+1), \rho^i(t+1), \boldsymbol{\mu}^i(t+1))$ exists and from the previous arguments $(\mathbf{x}^i(t+1), \boldsymbol{\mu}^i(t+1))$ solves (15), thus concluding the proof. \square

Remark 3.8 (*Alternative proof of Lemma 3.7*): Let $\boldsymbol{\mu}_s^i \geq 0$, $s \in \{1, \dots, S\}$ be (nonnegative) Lagrange multipliers associated to the inequality constraints of problem (3). Then, its (partial) Lagrangian can be written as

$$\begin{aligned} \mathcal{L}_4(\rho^i, \mathbf{x}^i, \boldsymbol{\mu}^i) = & \rho^i + \sum_{s=1}^S \boldsymbol{\mu}_s^i \left(g_s^i(\mathbf{x}_s^i) \right. \\ & \left. + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s - \rho^i \right) \end{aligned}$$

and collecting the multiplier ρ^i , we obtain

$$\begin{aligned} \mathcal{L}_4(\rho^i, \mathbf{x}^i, \boldsymbol{\mu}^i) = & \rho^i \left(1 - \sum_{s=1}^S \boldsymbol{\mu}_s^i \right) \\ & + \sum_{s=1}^S \boldsymbol{\mu}_s^i \left(g_s^i(\mathbf{x}_s^i) + \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right). \end{aligned}$$

The minimization of \mathcal{L}_4 with respect to ρ^i constrains the 1-norm of the dual variable $\boldsymbol{\mu}^i$ (i.e., $\mathbf{1}^\top \boldsymbol{\mu}^i = 1$). Then, minimizing the reminder over $\mathbf{x}^i \in X^i$ and maximizing the result over $\boldsymbol{\mu}^i \succeq 0$ gives problem (15). \square

We point out that in the previous lemma we have shown that the minimization in (3) turns out to be equivalent to performing step (S1). An important consequence of Lemma 3.6 is that each iteration of the algorithm can be in fact performed (since a primal-dual optimal solution pair of (3) exists). This is strictly related to the result of Lemma 3.5. In fact, the solvability of problem (3) is equivalent to the boundedness, at each t , of the subgradients of η . This is ensured, equivalently, by the compactness of the simplex constraint in (14).

The next corollary is a byproduct of the proof of Lemma 3.7.

Corollary 3.9: Let $\rho^i(t+1)$, for each $i \in \{1, \dots, N\}$, be the optimal cost of problem (3) with fixed values $\{\lambda^{ij}(t), \lambda^{ji}(t)\}_{j \in \mathcal{N}_i}$. Then, it holds that

$$\rho^i(t+1) = \eta^i(\{\lambda^{ij}(t), \lambda^{ji}(t)\}_{j \in \mathcal{N}_i}) \quad (21)$$

where η^i is defined in (11).

Proof: To prove the corollary, we first rewrite explicitly the definition of $\eta^i(\{\lambda^{ij}(t), \lambda^{ji}(t)\}_{j \in \mathcal{N}_i})$ given in (11), i.e.,

$$\begin{aligned} \eta^i(\{\lambda^{ij}(t), \lambda^{ji}(t)\}_{j \in \mathcal{N}_i}) = \\ \max_{\mathbf{1}^\top \boldsymbol{\mu}^i = 1, \boldsymbol{\mu}^i \succeq 0} \left(\left(\min_{\mathbf{x}^i \in X^i} \sum_{s=1}^S \boldsymbol{\mu}_s^i g_s^i(\mathbf{x}_s^i) \right) \right. \\ \left. + \boldsymbol{\mu}^{i^\top} \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t)) \right). \end{aligned} \quad (22)$$

Then, being $\rho_i(t)$ the optimal cost of problem (3), it is also the optimal cost of problem (20), which is equivalent to the right hand side of equation (19). The proof follows by noting that the expression of η^i in (22) is exactly the left hand side of (19) after rearranging some terms. \square

We are now ready to state the main result of the paper, namely the convergence of the DDPM distributed algorithm.

Theorem 3.10: Let $\{(\mathbf{x}^i(t), \rho^i(t))\}$, $i \in \{1, \dots, N\}$, be a sequence generated by the DDPM distributed algorithm, with $\gamma(t)$ satisfying Assumption B.3. Then, the following holds:

- (i) the sequence $\{\sum_{i=1}^N \rho^i(t)\}$ converges to the optimal cost P^* of problem (1), and
- (ii) every limit point of the primal sequence $\{\mathbf{x}^i(t)\}$, with $i \in \{1, \dots, N\}$, is an optimal (feasible) solution of (1).

Proof: We prove the theorem by combining all the results given in the previous lemmas.

First, for each $i \in \{1, \dots, N\}$, let $\{\boldsymbol{\mu}^i(t)\}$, and $\{\lambda^{ij}(t)\}$, $j \in \mathcal{N}_i$, be the auxiliary sequences defined in the DDPM distributed algorithm associated to $\{(\mathbf{x}^i(t), \rho^i(t))\}$. From Lemma 3.7 a primal-dual optimal solution pair $((\mathbf{x}^i(t+1), \rho^i(t+1)), \boldsymbol{\mu}^i(t+1))$ of (3) in fact exists (so that the algorithm is well-posed) and $(\mathbf{x}^i(t+1), \boldsymbol{\mu}^i(t+1))$ solves (15). Recalling that solving (15) is equivalent to solving (14), it follows that $\boldsymbol{\mu}^i(t+1)$ in the DDPM implements step (S1) of the dual subgradient (S1)-(S2). Noting that update (4) of λ^{ij} is exactly step (S2), it follows

that DDPM is an operative way to implement the dual subgradient algorithm (S1)-(S2). From Lemma 3.5 the algorithm converges in objective value, that is

$$\lim_{t \rightarrow \infty} \sum_{i=1}^N \eta^i(\{\lambda^{ij}(t), \lambda^{ji}(t)\}_{j \in \mathcal{N}_i}) = \eta^* = P^*,$$

where the second equality follows from Lemma 3.3. Then, we notice that from Corollary 3.9

$$\sum_{i=1}^N \rho^i(t) = \sum_{i=1}^N \eta^i(\{\lambda^{ij}(t), \lambda^{ji}(t)\}_{j \in \mathcal{N}_i}) \quad \forall t \geq 0,$$

so that $\lim_{t \rightarrow \infty} \sum_{i=1}^N \rho^i(t) = P^*$, thus concluding the proof of the first statement.

To prove the second statement, we show that every limit point of the (primal) sequence $\{\mathbf{x}^i(t)\}$, $i \in \{1, \dots, N\}$, is feasible and optimal for problem (1).

For analysis purposes, let us introduce the sequence $\{P(t)\}$ defined as

$$P(t) := \max_{s \in \{1, \dots, S\}} \sum_{i=1}^N g_s^i(\mathbf{x}_s^i(t)) \quad (23)$$

for each $t \geq 0$. Notice that $P(t)$ is also the cost of problem (2) associated to $[\mathbf{x}^1(t), \dots, \mathbf{x}^N(t), P(t)]$ and thus, by definition of optimality, satisfies

$$P^* \leq P(t) \quad (24)$$

for all $t \geq 0$.

By summing over $i \in \{1, \dots, N\}$ both sides of inequality constraints in (3), at each $t \geq 0$ the following holds

$$\sum_{i=1}^N \left(g_s^i(\mathbf{x}_s^i(t)) - \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t))_s \right) \leq \sum_{i=1}^N \rho^i(t). \quad (25)$$

Let us denote a_{ij} the (i, j) -th entry of the adjacency matrix associated to the undirected graph \mathcal{G} . Then, we can write

$$\begin{aligned} & \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t)) \\ &= \sum_{i=1}^N \sum_{j=1}^N a_{ij} (\lambda^{ij}(t) - \lambda^{ji}(t)) \\ &= \sum_{i=1}^N \sum_{j=1}^N a_{ij} \lambda^{ij}(t) - \sum_{i=1}^N \sum_{j=1}^N a_{ij} \lambda^{ji}(t). \end{aligned}$$

Since the graph \mathcal{G} is undirected $a_{ij} = a_{ji}$ for all $(i, j) \in \mathcal{E}$ and thus

$$\begin{aligned} & \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} (\lambda^{ij}(t) - \lambda^{ji}(t)) \\ &= \sum_{i=1}^N \sum_{j=1}^N a_{ij} \lambda^{ij}(t) - \sum_{i=1}^N \sum_{j=1}^N a_{ji} \lambda^{ji}(t) = 0. \end{aligned}$$

Hence, (25) reduces to

$$\sum_{i=1}^N g_s^i(\mathbf{x}_s^i(t)) \leq \sum_{i=1}^N \rho^i(t), \quad (26)$$

for all $s \in \{1, \dots, S\}$ and $t \geq 0$.

For all $i \in \{1, \dots, N\}$, since $\{\mathbf{x}^i(t)\}$ is a bounded sequence in X^i , then there exists a convergent sub-sequence $\{\mathbf{x}^i(t_n)\}$. Let $\bar{\mathbf{x}}^i$ be its limit point. Since each g_s^i is a (finite) convex function over \mathbb{R} , it is also continuous over any compact subset of \mathbb{R} and, taking the limit of (26), we can write

$$\begin{aligned} \lim_{n \rightarrow \infty} \sum_{i=1}^N g_s^i(\mathbf{x}_s^i(t_n)) &= \sum_{i=1}^N g_s^i\left(\lim_{n \rightarrow \infty} \mathbf{x}_s^i(t_n)\right) \\ &= \sum_{i=1}^N g_s^i(\bar{\mathbf{x}}_s^i) \leq \lim_{n \rightarrow \infty} \sum_{i=1}^N \rho^i(t_n) = P^* \end{aligned} \quad (27)$$

for $s \in \{1, \dots, S\}$, where the last equality follows from the first statement of the theorem. Since the sub-sequence $\{\mathbf{x}^i(t_n)\}$ is arbitrary, we have shown that every limit point $\bar{\mathbf{x}}^i$, $i \in \{1, \dots, N\}$, is feasible.

To show optimality, first notice that in light of conditions (24) and (26) the following holds

$$P^* \leq P(t) = \max_{s \in \{1, \dots, S\}} \sum_{i=1}^N g_s^i(\mathbf{x}_s^i(t)) \leq \sum_{i=1}^N \rho^i(t). \quad (28)$$

Therefore, taking any convergent sub-sequence $\{\mathbf{x}^i(t_n)\}$ (with limit point $\bar{\mathbf{x}}^i$) in (28), the limit as $n \rightarrow \infty$ satisfies

$$P^* \leq \lim_{n \rightarrow \infty} \left(\max_{s \in \{1, \dots, S\}} \sum_{i=1}^N g_s^i(\mathbf{x}_s^i(t_n)) \right) \leq \lim_{n \rightarrow \infty} \sum_{i=1}^N \rho^i(t_n) = P^*. \quad (29)$$

By noting that the maximization is over a finite set and recalling that g_s^i is continuous over any compact subset of \mathbb{R} , it follows

$$P^* \leq \max_{s \in \{1, \dots, S\}} \sum_{i=1}^N g_s^i(\bar{\mathbf{x}}_s^i) \leq P^* \quad (30)$$

proving that any limit point $\bar{\mathbf{x}}^i$, $i \in \{1, \dots, N\}$, is also optimal, thus concluding the proof. \square

IV. NUMERICAL SIMULATIONS

In this section we propose a numerical example in which we apply the proposed method to a network of Thermostatically Controlled Loads (TCLs) (such as air conditioners, heat pumps, electric water heaters), [21].

The dynamical model of the i -th device is given by

$$\dot{T}^i(\tau) = -\alpha (T^i(\tau) - T_{out}^i(\tau)) + \delta^i(\tau) + Qx^i(\tau), \quad (31)$$

where $T_i(\tau) \geq 0$ is the temperature, $\alpha > 0$ is a parameter depending on geometric and thermal characteristics, $T_{out}^i(\tau)$ is the air temperature outside the device, $\delta^i(\tau)$ represents a known time-varying forcing term onto the internal temperature of the device, $x^i(\tau) \in [0, 1]$ is the control input, and $Q > 0$ is a scaling factor.

We consider a discretized version of the system with constant input over the sampling interval $\Delta\tau$, i.e., $x^i(\tau) = \mathbf{x}_s^i$ for $\tau \in [s\Delta\tau, (s+1)\Delta\tau)$, and sampled state T_s^i ,

$$T_{s+1}^i = T_s^i e^{-\alpha\Delta\tau} + (1 - e^{-\alpha\Delta\tau}) \left(\frac{Q}{\alpha} \mathbf{x}_s^i + \frac{\delta_s^i}{\alpha} + T_{out,s}^i \right). \quad (32)$$

Moreover, we constrain the temperature to stay within a given interval $[T_{min}, T_{max}]$.

The constraints due to the dynamics and the bound on the temperature can be written as inequality constraints on the input in the form $A_i \mathbf{x}^i \leq b_i$, for each agent $i \in \{1, \dots, N\}$. To construct A_i and b_i , let us denote $\hat{A} = e^{-\alpha\Delta\tau}$ and $\hat{B} = 1 - e^{-\alpha\Delta\tau}$. We can compute the trajectory of T_s^i as a function of \mathbf{x}_s^i , δ_s^i , $T_{out,s}^i$ and T_0^i as follows. Let $\delta^i, \bar{T}^i, \bar{T}_{out}^i \in \mathbb{R}^S$ be vectors whose s -th element corresponds respectively to δ_s^i , T_s^i and $T_{out,s}^i$. Then, based on (32) it holds

$$\bar{T}^i = \underbrace{\begin{bmatrix} \hat{B} & 0 & \dots & 0 \\ \hat{A}\hat{B} & \hat{B} & \dots & 0 \\ \vdots & & & \\ \hat{A}^S \hat{B} & \hat{A}^{S-1} \hat{B} & \dots & \hat{B} \end{bmatrix}}_F \left(-\bar{T}_{out}^i + \frac{\delta^i}{\alpha} + \frac{Q}{\alpha} \mathbf{x}^i \right) + \underbrace{\begin{bmatrix} \hat{A} \\ \hat{A}^2 \\ \vdots \\ \hat{A}^S \end{bmatrix}}_G T_0^i.$$

Thus, the matrix A_i and the vector b_i turn out to be

$$A_i = \begin{bmatrix} \frac{Q}{\alpha} F \\ -\frac{Q}{\alpha} F \end{bmatrix}, \quad b_i = \begin{bmatrix} T_{max} \mathbf{1} - GT_0^i + F\bar{T}_{out}^i + F\frac{\delta^i}{\alpha} \\ -T_{min} \mathbf{1} + GT_0^i - F\bar{T}_{out}^i - F\frac{\delta^i}{\alpha} \end{bmatrix}.$$

We assume that the power consumption $g_s^i(\mathbf{x}_s^i)$ of the i -th device in the s -th slot $[s\Delta\tau, (s+1)\Delta\tau]$ is directly proportional to \mathbf{x}_s^i , i.e., $g_s^i(\mathbf{x}_s^i) = c^i \mathbf{x}_s^i$.

Thus, optimization problem (1) for this scenario is

$$\begin{aligned}
& \min_{\mathbf{x}^1, \dots, \mathbf{x}^N, P} P \\
& \text{subj. to } A_i \mathbf{x}^i \preceq b_i, \mathbf{x}^i \in [0, 1]^S, \quad i \in \{1, \dots, N\} \\
& \sum_{i=1}^N c^i \mathbf{x}_s^i \leq P, \quad s \in \{1, \dots, S\},
\end{aligned} \tag{33}$$

where A_i and b_i encode the constraints due to the discrete-time dynamics, the temperature constraint $T_s^i \in [T_{min}, T_{max}]$ and the known forcing term δ_s^i . Notice that the local constraint set is $X^i := \{\mathbf{x}^i \in \mathbb{R}^S \mid A_i \mathbf{x}^i \preceq b_i \text{ and } \mathbf{x}^i \in [0, 1]^S\}$.

We choose each δ_s^i to be constant for an interval of 5 slots and zero otherwise. The nonzero values are set in the central part of the entire simulation horizon $\{1, \dots, S\}$ by randomly shifting the center. Then, we randomly choose the heterogeneous power consumption coefficient $c^i \in \mathbb{R}$ of each device from a set of five values, drawn from a uniform distribution in $[1, 3]$. Finally, we consider $N = 20$ agents communicating according to an undirected connected Erdős-Rényi random graph \mathcal{G} with parameter 0.2. We consider a horizon of $S = 60$. Finally, we used a diminishing step-size sequence in the form $\gamma(t) = t^{-0.8}$, satisfying Assumption B.3.

In Figure 2 we show the evolution at each algorithm iteration t of the local objective functions $\rho^i(t)$, $i \in \{1, \dots, N\}$, (solid lines) which converge to stationary values. Moreover, we also plot their sum $\sum_{i=1}^N \rho^i(t)$ (dashed line) and the value $P(t)$ (dotted line), introduced in (23). As proven in Corollary 3.9, both of them asymptotically converge to the centralized optimal cost P^* of problem (33). It is worth noting that, at each iteration t , the curve $P(t)$ stays above the optimal value P^* and below the curve $\sum_{i=1}^N \rho^i(t)$, i.e., condition (28) is satisfied.

In Figure 3 the local solutions at the last algorithm iteration are depicted. We denote them \mathbf{x}^{i*} , $i \in \{1, \dots, N\}$, to highlight that they satisfy the cost optimality up to the required tolerance 10^{-3} . We also plot the resulting aggregate optimal consumption, i.e., $\sum_{i=1}^N c^i \mathbf{x}^{i*}$, which, as expected, in fact shaves off the power demand peak.

Moreover, the optimal local solutions satisfy the box constraint $[0, 1]$ for each slot $s \in \{1, \dots, S\}$. In fact, as we have proven, the algorithm converges in an interior point fashion, i.e., the local constraint at each node $i \in \{1, \dots, N\}$, is satisfied at all the algorithm iterations. As an example, in Figure 4 we depict the behavior of the components of $\mathbf{x}^1(t)$.

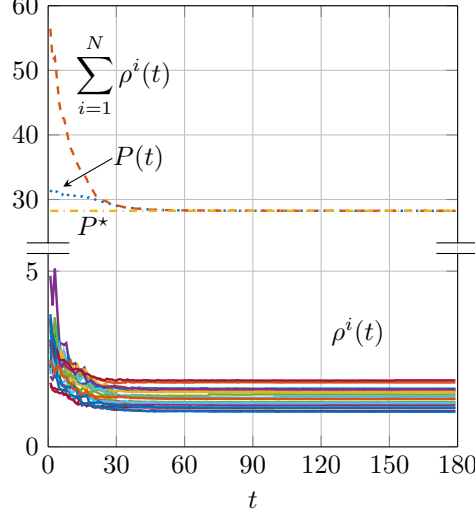


Fig. 2. Evolution of $\rho^i(t)$, $i \in \{1, \dots, N\}$, (solid lines), their sum $\sum_{i=1}^N \rho^i(t)$ (dashed line), $P(t)$ (dotted line) and (centralized) optimal cost P^* (dash-dotted line).

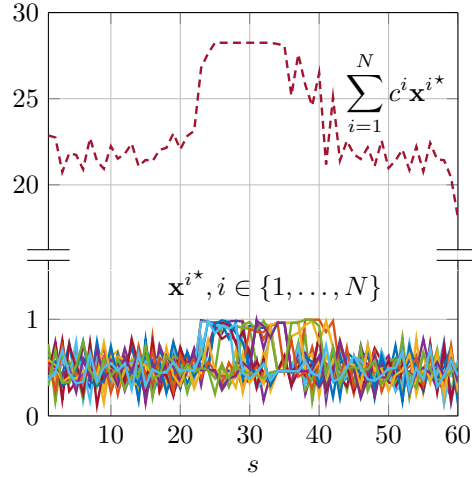


Fig. 3. Profile of optimal solutions \mathbf{x}^{i*} (solid lines), and $\sum_{i=1}^N c^i \mathbf{x}_s^{i*}$ (dashed line) over the horizon $\{1, \dots, S\}$.

In Figure 5 (left) we show, the violation of the coupling constraints, for all $s \in \{1, \dots, S\}$ at each iteration t . As expected, the violations asymptotically go to nonnegative values, consistently with the asymptotic primal feasibility proven in the previous section.

In Figure 5 (right) the difference $\sum_{i=1}^N (g_s^i(\mathbf{x}_s^i(t)) - \rho^i(t))$ is also shown, which is always nonnegative consistently with equation (26).

Finally, in Figure 6 it is shown the convergence rate of the distributed algorithm, i.e., the

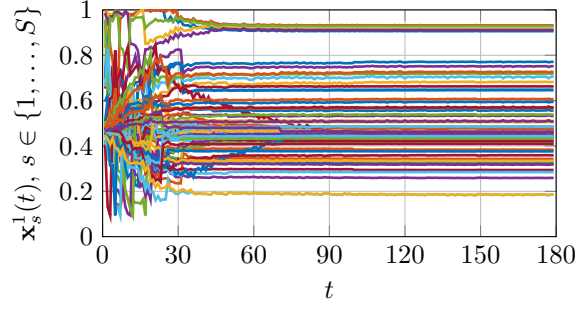


Fig. 4. Evolution of $\mathbf{x}_s^1(t)$, $s \in \{1, \dots, S\}$.

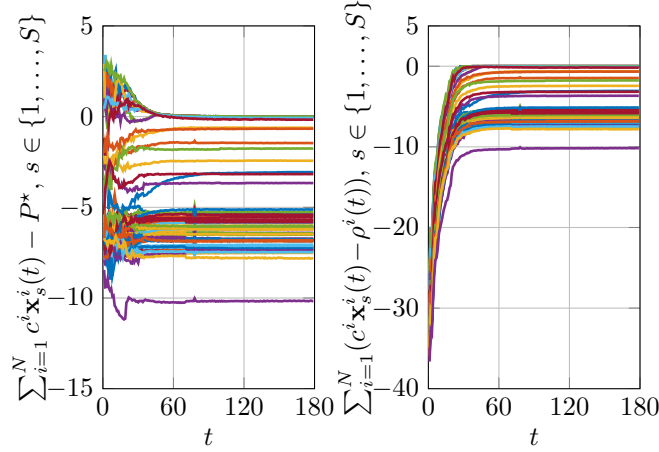


Fig. 5. Evolution of primal violations of solutions $\mathbf{x}^i(t)$, $i \in \{1, \dots, N\}$.

difference between the centralized optimal cost P^* and the sum of the local costs $\sum_{i=1}^N \rho^i(t)$, in logarithmic scale. It can be seen that the proposed algorithm converges to the optimal cost with a sublinear rate $O(1/\sqrt{t})$ as expected for a subgradient method. Notice that the cost error is not monotone since the subgradient algorithm is not a descent method.

V. CONCLUSIONS

In this paper we have introduced a novel distributed min-max optimization framework motivated by peak minimization problems in Demand Side Management. Standard distributed optimization algorithms cannot be applied to this problem set-up due to a highly nontrivial coupling in the objective function and in the constraints. We proposed a distributed algorithm based on the combination of duality methods and properties from min-max optimization. Specifically, by means of duality theory, a series of equivalent problems are set-up, which lead to a separable and

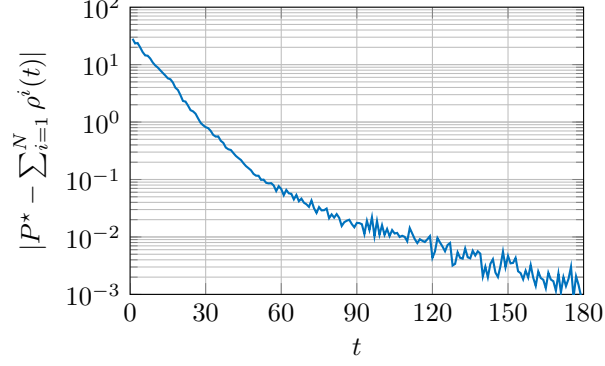


Fig. 6. Evolution of the cost error, in logarithmic scale.

sparse optimization problem. A subgradient method applied to the resulting problem results into a distributed algorithm by suitably applying properties from min-max optimization. Despite the complex derivation, the algorithm has a very simple structure at each node. Theoretical results are corroborated by a numerical example on peak minimization in Demand Side Management.

APPENDIX

A. Optimization and Duality

Consider a constrained optimization problem, addressed as primal problem, having the form

$$\begin{aligned} \min_{z \in Z} f(z) \\ \text{subj. to } g(z) \preceq 0 \end{aligned} \quad (\text{A.34})$$

where $Z \subseteq \mathbb{R}^N$ is a convex and compact set, $f : \mathbb{R}^N \rightarrow \mathbb{R}$ is a convex function and $g : \mathbb{R}^N \rightarrow \mathbb{R}^S$ is such that each component $g_s : \mathbb{R}^N \rightarrow \mathbb{R}$, $s \in \{1, \dots, S\}$, is a convex function.

The following optimization problem

$$\begin{aligned} \max_{\mu} q(\mu) \\ \text{subj. to } \mu \succeq 0 \end{aligned} \quad (\text{A.35})$$

is called the dual of problem (A.34), where $q : \mathbb{R}^S \rightarrow \mathbb{R}$ is obtained by minimizing with respect to $z \in Z$ the Lagrangian function $\mathcal{L}(z, \mu) := f(z) + \mu^\top g(z)$, i.e., $q(\mu) = \min_{z \in Z} \mathcal{L}(z, \mu)$. Problem (A.35) is well posed since the domain of q is convex and q is concave on its domain.

It can be shown that the following inequality holds

$$\inf_{z \in Z} \sup_{\mu \succeq 0} \mathcal{L}(z, \mu) \geq \sup_{\mu \succeq 0} \inf_{z \in X} \mathcal{L}(z, \mu), \quad (\text{A.36})$$

which is called weak duality. When in (A.36) the equality holds, then we say that strong duality holds and, thus, solving the primal problem (A.34) is equivalent to solving its dual formulation (A.35). In this case the right-hand-side problem in (A.36) is referred to as *saddle-point problem* of (A.34).

Definition A.1: A pair (z^*, μ^*) is called a primal-dual optimal solution of problem (A.34) if $z^* \in Z$ and $\mu^* \succeq 0$, and (z^*, μ^*) is a saddle point of the Lagrangian, i.e.,

$$\mathcal{L}(z^*, \mu) \leq \mathcal{L}(z^*, \mu^*) \leq \mathcal{L}(z, \mu^*)$$

for all $z \in Z$ and $\mu \succeq 0$. □

A more general min-max property can be stated. Let $Z \subseteq \mathbb{R}^N$ and $W \subseteq \mathbb{R}^S$ be nonempty convex sets. Let $\phi : Z \times W \rightarrow \mathbb{R}$, then the following inequality

$$\inf_{z \in Z} \sup_{w \in W} \phi(z, w) \geq \sup_{w \in W} \inf_{z \in Z} \phi(z, w)$$

holds true and is called the *max-min* inequality. When the equality holds, then we say that ϕ , Z and W satisfy the *strong max-min* property or the *saddle-point* property.

The following theorem gives a sufficient condition for the strong max-min property to hold.

Proposition A.2 ([22, Propositions 4.3]): Let ϕ be such that (i) $\phi(\cdot, w) : Z \rightarrow \mathbb{R}$ is convex and closed for each $w \in W$, and (ii) $-\phi(z, \cdot) : W \rightarrow \mathbb{R}$ is convex and closed for each $z \in Z$. Assume further that W and Z are convex compact sets. Then

$$\sup_{w \in W} \inf_{z \in Z} \phi(z, w) = \inf_{z \in Z} \sup_{w \in W} \phi(z, w)$$

and the set of saddle points is nonempty and compact. □

B. Subgradient Method

Consider the following (constrained) optimization problem

$$\min_{z \in Z} f(z) \tag{B.37}$$

with $f : \mathbb{R}^N \rightarrow \mathbb{R}$ a convex function and $Z \subseteq \mathbb{R}^N$ a closed, convex set.

A vector $\tilde{\nabla} f(z) \in \mathbb{R}^N$ is called a subgradient of the convex function f at $z \in \mathbb{R}^N$ if $f(y) \geq f(z) + \tilde{\nabla} f(z)(y - z)$ for all $y \in \mathbb{R}^N$. The (projected) subgradient method is the iterative algorithm given by

$$z(t+1) = P_Z \left(z(t) - \gamma(t) \tilde{\nabla} f(z(t)) \right) \tag{B.38}$$

where $t \geq 0$ denotes the iteration index, $\gamma(t)$ is the step-size, $\tilde{\nabla}f(z(t))$ denotes a subgradient of f at $z(t)$, and $P_Z(\cdot)$ is the Euclidean projection onto Z .

The following standard assumption is usually needed to guarantee convergence of the subgradient method.

Assumption B.3: The sequence $\{\gamma(t)\}$, with $\gamma(t) \geq 0$ for all $t \geq 0$, satisfies the diminishing condition

$$\lim_{t \rightarrow \infty} \gamma(t) = 0, \quad \sum_{t=1}^{\infty} \gamma(t) = \infty, \quad \sum_{t=1}^{\infty} \gamma(t)^2 < \infty. \quad \square$$

The following proposition formally states the convergence of the subgradient method.

Proposition B.4 ([23, Proposition 3.2.6]): Assume that the subgradients $\tilde{\nabla}f(z)$ are bounded for all $z \in Z$ and the set of optimal solutions is nonempty. Let the step-size $\{\gamma(t)\}$ satisfy the diminishing condition in Assumption B.3. Then the subgradient method in (B.38) applied to problem (B.37) converges in objective value and sequence $\{z(t)\}$ converges to an optimal solution. \square

REFERENCES

- [1] I. Notarnicola, M. Franceschelli, and G. Notarstefano, “A duality-based approach for distributed min-max optimization with application to demand side management,” in *IEEE 55th Conference on Decision and Control (CDC)*, 2016.
- [2] M. Alizadeh, X. Li, Z. Wang, A. Scaglione, and R. Melton, “Demand-side management in the smart grid: Information processing for the power switch,” *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 55–67, 2012.
- [3] A.-H. Mohsenian-Rad, V. W. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, “Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid,” *IEEE Transactions on Smart Grid*, vol. 1, no. 3, pp. 320–331, 2010.
- [4] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa, “Demand-side management via distributed energy generation and storage optimization,” *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 866–876, 2013.
- [5] D. P. Palomar and M. Chiang, “A tutorial on decomposition methods for network utility maximization,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1439–1451, 2006.
- [6] B. Yang and M. Johansson, “Distributed optimization and games: A tutorial overview,” in *Networked Control Systems*. Springer, 2010, pp. 109–148.
- [7] M. Zhu and S. Martínez, “On distributed convex optimization under inequality and equality constraints,” *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 151–164, 2012.
- [8] T.-H. Chang, A. Nedić, and A. Scaglione, “Distributed constrained optimization by consensus-based primal-dual perturbation method,” *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1524–1538, 2014.
- [9] A. Nedić and A. Ozdaglar, “Approximate primal solutions and rate analysis for dual subgradient methods,” *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1757–1780, 2009.
- [10] J. Lu and M. Johansson, “Convergence analysis of primal solutions in dual first-order methods,” in *IEEE 52nd Conference on Decision and Control (CDC)*, 2013, pp. 6861–6867.

- [11] A. Simonetto and H. Jamali-Rad, “Primal recovery from consensus-based dual decomposition for distributed convex optimization,” *Journal of Optimization Theory and Applications*, vol. 168, no. 1, pp. 172–197, 2016.
- [12] A. Falsone, K. Margellos, S. Garatti, and M. Prandini, “Dual decomposition and proximal minimization for multi-agent distributed optimization with coupling constraints,” *arXiv preprint arXiv:1607.00600*, 2016.
- [13] A. Nedić and A. Ozdaglar, “Subgradient methods for saddle-point problems,” *Journal of optimization theory and applications*, vol. 142, no. 1, pp. 205–228, 2009.
- [14] K. Srivastava, A. Nedić, and D. Stipanović, “Distributed min-max optimization in networks,” in *IEEE 17th International Conference on Digital Signal Processing (DSP)*, 2011, pp. 1–8.
- [15] G. Notarstefano and F. Bullo, “Distributed abstract optimization via constraints consensus: Theory and applications,” *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2247–2261, 2011.
- [16] M. Bürger, G. Notarstefano, and F. Allgöwer, “A polyhedral approximation framework for convex and robust distributed optimization,” *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 384–395, 2014.
- [17] D. Mateos-Núñez and J. Cortés, “Distributed subgradient methods for saddle-point problems,” in *IEEE 54th Conference on Decision and Control (CDC)*, 2015, pp. 5462–5467.
- [18] A. Simonetto, T. Keviczky, and M. Johansson, “A regularized saddle-point algorithm for networked optimization with resource allocation constraints,” in *IEEE 51st Conference on Decision and Control (CDC)*, 2012, pp. 7476–7481.
- [19] A. Koppel, F. Y. Jakubiec, and A. Ribeiro, “Regret bounds of a distributed saddle point algorithm,” in *IEEE 40th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 2969–2973.
- [20] D. P. Bertsekas, *Nonlinear programming*. Athena scientific, 1999.
- [21] M. Alizadeh, A. Scaglione, A. Applebaum, G. Kesidis, and K. Levitt, “Reduced-order load models for large populations of flexible appliances,” *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 1758–1774, 2015.
- [22] D. P. Bertsekas, “Min common/max crossing duality: A geometric view of conjugacy in convex optimization,” *Lab. for Information and Decision Systems, MIT, Tech. Rep. Report LIDS-P-2796*, 2009.
- [23] —, *Convex Optimization Algorithms*. Athena Scientific, 2015.



Ivano Notarnicola has been a Ph.D. student in Engineering of Complex Systems at the Università del Salento (Lecce, Italy) since November 2014. He received the Laurea degree “summa cum laude” in Computer Engineering from the Università del Salento in 2014. He was a visiting scholar at the University of Stuttgart from March to June 2014. His research interests include distributed optimization and randomized algorithms.



Mauro Franceschelli (M'11) was born in 1983. He received the B.S. and M.S. degree in Electronic Engineering “cum laude” in 2005 and 2007 from the University of Cagliari, Italy. He received the PhD from the same University in 2011 where he continued its research collaboration with a postdoc until 2015. He has spent visiting periods at the Royal Institute of Technology (KTH), the Georgia Institute of Technology (GaTech), the University of California at Santa Barbara (UCSB) and Xidian University, Xi'an, China. Since 2015 he is Assistant Professor (RTD) at the Department of Electrical and Electronic Engineering, University of Cagliari, Italy, as recipient of a grant founded by the Italian Ministry of Education, University and Research (MIUR) under the 2014 call “Scientific Independence of Young Researchers” (SIR). He has published more than 50 papers in top international journals and conference proceedings. His research interests include consensus problems, gossip algorithms, multi-agent systems, multi-robot systems, distributed optimization and electric demand side management.



Giuseppe Notarstefano is Associate Professor at the Università del Salento (Lecce, Italy), where he was Assistant Professor (Ricercatore) from February 2007 to May 2016. He received the Laurea degree “summa cum laude” in Electronics Engineering from the Università di Pisa in 2003 and the Ph.D. degree in Automation and Operation Research from the Università di Padova in April 2007. He has been visiting scholar at the University of Stuttgart, University of California Santa Barbara and University of Colorado Boulder. His research interests include distributed optimization, cooperative control in complex networks, applied nonlinear optimal control, and trajectory optimization and maneuvering of aerial and car vehicles. He serves as an Associate Editor in the Conference Editorial Board of the IEEE Control Systems Society and for other IEEE and IFAC conferences. He coordinated the VI-RTUS team winning the International Student Competition Virtual Formula 2012. He is recipient of an ERC Starting Grant 2014.