Accelerated Multi-Agent Optimization Method over Stochastic Networks

Wicak Ananduta, Carlos Ocampo-Martinez, and Angelia Nedić

Abstract— We propose a distributed method to solve a multi-agent optimization problem with strongly convex cost function and equality coupling constraints. The method is based on Nesterov's accelerated gradient approach and works over stochastically time-varying communication networks. We consider the standard assumptions of Nesterov's method and show that the sequence of the expected dual values converge toward the optimal value with the rate of $\mathcal{O}(1/k^2)$. Furthermore, we provide a simulation study of solving an optimal power flow problem with a well-known benchmark case.

Index Terms— multi-agent optimization, distributed method, accelerated gradient method, distributed optimal power flow problem

I. INTRODUCTION

The advancement on information, computation and communication technologies promotes the deployment of distributed approaches to solve complex large-scale problems, e.g., in power networks [1], [2] and water networks [3]. On one hand, such approaches offer flexibility and scalability. On the other hand, they require more complex design than the centralized counterpart as multiple computational units must cooperate and communicate among each other.

In this paper, we deal with a multi-agent optimization problem, in which the cost function is a summation of a strongly convex cost functions. Moreover, the problem has equality coupling constraints. This formulation is mainly motivated from optimal power flow (OPF) problems of large-scale power networks [1] and resource allocation problems [2], [4]. Furthermore, the problem can also be considered as a subclass of extended monotropic problems [5].

We solve the problem in a distributed manner through its dual to deal with the coupling constraints. Particularly, we develop the method based on Nesterov's accelerated gradient method [6], [7], which is an accelerated first-order approach, with the rate of $\mathcal{O}(1/k^2)$. This accelerated method has been used to develop a fast distributed gradient method to solve network utility maximization problems [8], a fast alternating direction method of multipliers (ADMM) for a certain class of problems with strongly convex cost function [9], and distributed model predictive controllers [10], among others.

However, different from the aforementioned papers, one feature of the system that we particularly pay attention to is the time-varying nature of the communication network, over which the agents exchange information. Specifically

W. Ananduta is with the Delft Center of Systems and Control (DCSC), TU Delft, the Netherlands. C. Ocampo-Martinez is with Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain. A. Nedić is with School of Electrical, Computer and Energy Engineering, Arizona State University. E-mail addresses: `w.ananduta@tudelft.nl`, `carlos.ocampo@upc.edu`, `angelia.nedich@asu.edu`.

here, we assume that the network is stochastically time-varying and this assumption can model communication failures that might occur in large-scale systems. Similar setup on communication networks can be found in [11]–[14], which develop unaccelerated first-order methods, and [15], [16], which propose a Nesterov-like fast gradient method for distributed optimization problem with a common decision variable. Nevertheless, whereas the former four papers do not consider an accelerated method, the latter ones deal with a different problem and work directly in the primal space. Note that different models of time-varying communication networks have also been considered, as in [17]–[19].

To summarize, the main contribution of this paper is an accelerated first-order distributed method for a multi-agent optimization problem, which works over stochastic communication networks. As a fully distributed algorithm, the parameter design and iterations only need local information, i.e., neighbor-to-neighbor communication. Furthermore, since the method is based on Nesterov's accelerated approach, it enjoys the convergence rate of $\mathcal{O}(1/k^2)$ on the expected dual value, as shown in the convergence analysis.

The paper is structured as follows. Section II provides the problem setup and the cosidered model of time-varying communication networks. Afterward, Section III presents the proposed distributed method along with its convergence statement. Then, in Section IV, we show the convergence analysis of the proposed method. Furthermore, we also showcase the performance of the proposed method to solve an intra-day OPF problem for a well-known benchmark case in Section V. Finally, Section VI concludes the paper by providing some remarks and discussions about future work.

Notation and properties

The set of real numbers is denoted by $\mathbb{R}$. For any $a \in \mathbb{R}$, $\mathbb{R}_{\geq a}$ denotes $\{b \in \mathbb{R} : b \geq a\}$. The inner product of vectors $x, y \in \mathbb{R}^n$ is denoted by $\langle x, y \rangle$, whereas the Euclidean vector norm and the induced matrix norm are denoted by $\|\cdot\|$. The operator $\text{col}\{\cdot\}$ stacks the arguments column-wise. We use 0_n to denote zero vector with dimension n. When the dimension is clear from the context, we may omit the subscript. Furthermore, the following properties will be used in the convergence analysis.

Property 1 (Strong convexity): A differentiable function $f(x) : \mathbb{R}^n \to \mathbb{R}$ is strongly convex, if for any $x, y \in \mathbb{R}^n$ it holds that

$$\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq \sigma \|y - x\|^2,$$

where σ is the strong convexity constant. $\qquad\square$

Property 2 (Lipschitz smoothness): A function $f(x)$: $\mathbb{R}^n \to \mathbb{R}$ is continuously differentiable with Lipschitz continuous gradient, if for any $x, y \in \mathbb{R}^n$ it holds that

$$\|\nabla f(y) - \nabla f(x)\| \leq L\|y - x\|,$$

where L denotes the Lipschitz constant. $\square$

II. PROBLEM SETUP

A. Multi-agent optimization problem

We consider a multi-agent system, where the set of agents is denoted by $\mathcal{N} := \{1, 2, \ldots, N\}$. The agents want to cooperatively solve an optimization problem in the following form:

$$\operatorname*{minimize}_{u_i \in \mathcal{U}_i, \forall i \in \mathcal{N}} \quad \sum_{i=1}^{N} f_i(u_i) \tag{1a}$$

$$\text{s.t.} \quad G_i^i u_i + \sum_{j \in \mathcal{N}_i} G_i^j u_j = g_i, \quad \forall i \in \mathcal{N}, \tag{1b}$$

where $u_i \in \mathbb{R}^{n_i}$ and $\mathcal{U}_i \in \mathbb{R}^{n_i}$ denote the decision vector and the local set constraint of agent i, respectively. In (1a), each cost function $f_i(u_i)$ is associated to agent i. Moreover, each equality in (1b), with the non-zero matrix $G_i^j \in \mathbb{R}^{m_i \times n_j}$, for each $j \in \mathcal{N}_i \cup \{i\}$ and $i \in \mathcal{N}$, and $g_i \in \mathbb{R}^{m_i}$, is assigned to agent i and couples agent i with some other agents, i.e., $j \in \mathcal{N}_i \subseteq \mathcal{N}$. Based on the formulation of the coupling constraints in (1b), we can represent the system as a directed graph, denoted by $\mathcal{S} = (\mathcal{N}, \mathcal{V})$, where $\mathcal{V}$ denotes the set of links that represents how each agent influences the coupling constraint (1b) of other agents. Specifically, the link $(j, i) \in \mathcal{V}$ implies that u_j appears on the coupling constraint of agent i, i.e., $j \in \mathcal{N}_i$. Therefore, we can say that $\mathcal{N}_i$ is the set of in-neighbors of agent i. On the other hand, we also introduce the set of out-neighbors, denoted by $\mathcal{M}_i$, i.e., $\mathcal{M}_i = \{j \in \mathcal{N} : (i, j) \in \mathcal{V}\}$. Furthermore, we define $i \in \mathcal{M}_i$ and, in general, $\mathcal{M}_i$ may not be equal to $\mathcal{N}_i \cup \{i\}$ (see Figure 1).

Problem (1) is a subclass of the extended monotropic problem [5]. Resource allocation problems [2], [4] can also be formulated as in (1). A particular practical problem of interest, which can be represented by (1), is the direct current (DC) OPF problem [1], where the decision vectors u_i might consist of the real powers and phase angle, whereas (1b) represents the DC approximation of the power flow equations. Note that, in the DC-OPF problem, $\mathcal{M}_i = \mathcal{N}_i \cup \{i\}$.

Now, we consider the following assumptions hold.

Assumption 1: The function $f_i : \mathbb{R}^{n_i} \to \mathbb{R}$, for each $i \in \mathcal{N}$, is differentiable and strongly convex with strong convexity parameter denoted by σ_i. $\square$

Assumption 2: The local set $\mathcal{U}_i$, for each $i \in \mathcal{N}$, is compact and convex. $\square$

Assumption 3: The feasible set of Problem (1) is non-empty. $\square$

Assumptions 1 and 2 are rather restrictive, however, commonly used in the applications considered, i.e., OPF and resource allocation problems. Moreover, these assumptions allow us to apply Nesterov's accelerated gradient method to solve the dual problem of (1), as these assumptions result

Set of coupling constraints

$$G_1^1 u_1 + G_1^2 u_2 = g_1$$

$$G_2^2 u_2 + G_2^3 u_3 = g_2$$

$$G_3^3 u_3 + G_3^1 u_1 = g_3$$
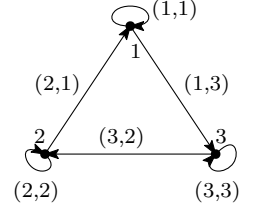
Graph representation

Fig. 1. A small network of three agents. Notice that $\mathcal{N}_1 = \{2\}$ and $\mathcal{M}_1 = \{1, 3\}$.

in a dual function with Lipschitz continuous gradient. This statement is elaborated further in Section IV. Furthermore, Assumption 3 is considered to ensure that the proposed algorithm can find a solution to Problem (1).

B. Stochastic communication networks

The aim of this work is to design a distributed optimization algorithm that solves Problem (1). As a distributed method, the algorithm requires each agent to communicate with other agents over a communication network, which we suppose to be time-varying. Precisely, the communication network is represented by the undirected graph $\mathcal{G}(k) = (\mathcal{N}, \mathcal{L}(k))$, where $\mathcal{L}(k) \subseteq \mathcal{N} \times \mathcal{N}$ denotes the set of communication links that may vary over iteration k, i.e., $\{i, j\} \in \mathcal{L}(k)$ implies that agents i and j can communicate at iteration k. Thus, we denote by $\mathcal{E}_i(k)$ the set of agents that can exchange information with agent i, i.e., $\mathcal{E}_i(k) = \{j \in \mathcal{N} : \{i, j\} \in \mathcal{L}(k)\}$. Furthermore, we consider the activation of communication links as a random process and the following assumption holds.

Assumption 4: The set $\mathcal{L}(k)$ is a random variable that is independent and identically distributed across iterations. Furthermore, any communication link of neighboring agents is active with a positive probability denoted by $\beta_{\{i,j\}}$, i.e., $\mathbb{P}(\{i, j\} \in \mathcal{L}(k)) = \beta_{\{i,j\}} > 0$, for $\{i, j\} \in \{\{i', j'\} \in \mathcal{N} \times \mathcal{N} : j' \in \mathcal{N}_{i'}, i' \in \mathcal{N}\}$. Additionally, $\beta_{\{i,i\}} = 1$, for all $i \in \mathcal{N}$. $\square$

Assumption 4 implies that the probability that agent i can receive information from all its in-neighbors $j \in \mathcal{N}_i$ at the same iteration k is positive. Let α_i denote this probability, thus we have that $\alpha_i = \prod_{j \in \mathcal{N}_i} \beta_{\{i,j\}}$.

III. PROPOSED METHOD

In this section, we propose a distributed method to solve Problem (1) over stochastic communication networks. The proposed method actually solves the dual problem associated to (1) and is based on Nesterov's accelerated gradient approach [6], [7].

To that end, let $\lambda_i \in \mathbb{R}^{m_i}$ denote the Lagrange multiplier associated to (1b), for each $i \in \mathcal{N}$, and $\lambda = \operatorname{col}\{\lambda_i, i \in \mathcal{N}\}$. Thus, we define the dual function, associated to (1) and denoted by $q(\lambda)$, as follows:

$$q(\lambda) = \sum_{i \in \mathcal{N}} q_i(\lambda^i), \tag{2}$$

Algorithm 1 Distributed accelerated method

Initialization (for each $i \in \mathcal{N}$)
Set $\theta(1) = 1$ and $\hat{\lambda}_i(1) = \lambda_i(0) = 0$
Iteration (for each $i \in \mathcal{N}$, $k \geq 1$)

1) Compute $u_i(k)$:

$$u_i(k) = \arg \min_{u_i \in \mathcal{U}_i} f_i(u_i) + \sum_{j \in \mathcal{M}_i} \langle G_j^{i\top} \hat{\lambda}_j(k), u_i \rangle \quad (5)$$

2) Send $G_j^i u_i(k)$ to out-neighbors $j \in \mathcal{M}_i$ and receive $G_i^j u_j(k)$ from the in-neighbors $j \in \mathcal{N}_i$
3) Compute $\lambda_i(k)$:

$$\lambda_i(k) = \hat{\lambda}_i(k) + \eta_i \left(G_i^i u_i(k) + \sum_{j \in \mathcal{N}_i} G_i^j u_j(k) - g_i \right) \quad (6)$$

4) Compute $\theta(k+1) = \frac{1 + \sqrt{1 + 4\theta^2(k)}}{2}$
5) Compute $\hat{\lambda}_i(k+1)$:

$$\hat{\lambda}_i(k+1) = \lambda_i(k) + \frac{\theta(k) - 1}{\theta(k+1)} (\lambda_i(k) - \lambda_i(k-1)) \quad (7)$$

6) Send $\hat{\lambda}_i(k+1)$ to in-neighbors $j \in \mathcal{N}_i$ and receive $\hat{\lambda}_j(k+1)$ from the out-neighbors $j \in \mathcal{M}_i$

---

where

$$q_i(\lambda^i) = \min_{u_i \in \mathcal{U}_i} \left\{ f_i(u_i) - \langle \lambda_i, g_i \rangle + \sum_{j \in \mathcal{M}_i} \langle G_j^{i\top} \lambda_j, u_i \rangle \right\}. \quad (3)$$

Note that λ^i denotes all Lagrange multipliers associated to the coupling constraints that involve agent i, i.e., $\lambda^i = \operatorname{col}\{\lambda_j, j \in \mathcal{M}_i\}$. We will then solve the dual problem:

$$\text{maximize } q(\lambda), \quad (4)$$

by adapting Nesterov's accelerated gradient method such that it works over stochastically time-varying communication networks (c.f. Section II-B). Note that, due to Assumptions 1-3, the strong duality holds [20, Proposition 5.2.1].

Hence, first we state the distributed method based on Nesterov's accelerated gradient approach without considering stochastic communication networks, i.e., the information required to perform the updates is always available. The method is shown in Algorithm 1. For a detailed design procedure of Nesterov's accelerated method, the reader might check [7], [8]. The main steps in the iteration of Nesterov's accelerated approach can be seen in Steps 4 and 5 where an interpolated point of each Lagrange multiplier λ_i (denoted by $\hat{\lambda}_i$) is computed. As a distributed method, these steps are carried out by each agent. Furthermore, the step-size of the gradient ascent in (6), denoted by η_i, is a local variable that must be chosen appropriately (c.f. Theorem 1). Finally, note that, in (5), u_i is updated by solving a local minimization derived from (3) and based on the interpolated points of the Lagrange multipliers from the out-neighbors,

---

Algorithm 2 Distributed accelerated method over stochastic networks

Initialization (for each $i \in \mathcal{N}$)
Set $\theta(1) = 1$, $\lambda_i(0) = 0$, and $\hat{\xi}_j^i(1) = \xi_j^i(0) = 0$, for all $j \in \mathcal{M}_i$
Iteration (for each $i \in \mathcal{N}$, $k \geq 1$): with random realization of $\mathcal{L}(k)$

1) Compute $u_i(k)$:

$$u_i(k) = \arg \min_{u_i \in \mathcal{U}_i} f_i(u_i) + \sum_{j \in \mathcal{M}_i} \langle G_j^{i\top} \hat{\xi}_j^i(k), u_i \rangle \quad (8)$$

2) Send $G_j^i u_i(k)$ to out-neighbors $j \in \mathcal{E}_i(k) \cap \mathcal{M}_i$ and receive $G_i^j u_j(k)$ from in-neighbors $j \in \mathcal{E}_i(k) \cap \mathcal{N}_i$
3) Compute $\lambda_i(k)$:

$$\lambda_i(k) = \begin{cases} \hat{\xi}_i^i(k) + \eta_i \left(G_i^i u_i(k) + \sum_{j \in \mathcal{N}_i} G_i^j u_j(k) - g_i \right), \\ \qquad \text{if } \mathcal{N}_i \subseteq \mathcal{E}_i(k) \\ \hat{\xi}_i^i(k), \quad \text{otherwise} \end{cases} \quad (9)$$

4) Send $\lambda_i(k)$ to in-neighbors $j \in \mathcal{E}_i(k) \cap \mathcal{N}_i$ and receive $\lambda_j(k)$ from out-neighbors $j \in \mathcal{E}_i(k) \cap \mathcal{M}_i$
5) Update $\xi_j^i(k)$, for all $j \in \mathcal{M}_i$:

$$\xi_j^i(k) = \begin{cases} \lambda_j(k), \text{ for } j \in \mathcal{M}_i \cap \mathcal{E}_i(k), \\ \hat{\xi}_j^i(k), \quad \text{otherwise} \end{cases} \quad (10)$$

6) Compute $\theta(k+1) = \frac{1 + \sqrt{1 + 4\theta^2(k)}}{2}$
7) Compute $\hat{\xi}_j^i(k+1)$, for all $j \in \mathcal{M}_i$:

$$\hat{\xi}_j^i(k+1) = \xi_j^i(k) + \frac{\theta(k) - 1}{\theta(k+1)} (\xi_j^i(k) - \xi_j^i(k-1)) \quad (11)$$

---

i.e., $\hat{\lambda}^i = \operatorname{col}\{\hat{\lambda}_j, j \in \mathcal{M}_i\}$. Due to Assumptions 1 and 2, the local minimization in Step 1 admits a unique solution.

Now, we are ready to state the proposed method, which works over stochastic communication networks. The method is shown in Algorithm 2. We adjust the gradient step update (Step 3) in order to take into account the time-varying nature of the communication network. As can be seen in Step 3, λ_i is only updated with the gradient step when agent i receives new information from all in-neighbors in $\mathcal{N}_i$. Furthermore, the required Lagrange multipliers from the other agents $j \in \mathcal{M}_i$ are tracked by agent i using the auxiliary vector $\xi^i = \operatorname{col}\{\xi_j^i, j \in \mathcal{M}_i\}$, where each ξ_j^i is updated in (10). Additionally, each agent i must compute the interpolated point of ξ_j^i, denoted by $\hat{\xi}_j^i$ in (11). This step is different than the steps in Algorithm 1, where the exchanged information is actually the interpolated point $\hat{\lambda}_i$.

The outcome of Algorithm 2, which is the main result of this work, is stated as the following theorem.

Theorem 1: Let Assumptions 1-4 hold and the sequence $\lambda(k)$ be generated by Algorithm 2 with $\eta_i \in (0, 1/L_i]$, where L_i is defined as follows:

$$L_i = \sum_{j \in \mathcal{N}_i \cup \{i\}} \frac{\|G^j\|^2}{\sigma_j}, \quad (12)$$

in which $G^j = \text{col}\{G_i^j, i \in \mathcal{M}_j\}$ and σ_j is the strong convexity constant of $f_j(u_j)$. Furthermore, let $q(\lambda)$ be defined by (2) and $\lambda^\star$ be an optimal solution of the dual problem (4). Then,

1) It holds that

$$\mathbb{E}\left(q(\lambda^\star) - q(\lambda(k))\right) \leq \frac{C}{(k+1)^2}, \qquad (13)$$

where C is a non-negative constant.

2) Hence, it also holds that

$$\lim_{k \to \infty} \mathbb{E}\left(q(\lambda^\star) - q(\lambda(k))\right) = 0, \qquad (14)$$

almost surely. $\qquad \square$

Theorem 1 shows that the expected dual values converge to the optimal dual value with the rate of $\mathcal{O}(1/k^2)$. Furthermore, the choice of parameter η_i, for each agent $i \in \mathcal{N}$, which is sufficient to achieve convergence, can be obtained locally, i.e., agent i only requires some information from its in-neighbors in $\mathcal{N}_i$ (see (12)).

IV. CONVERGENCE ANALYSIS

First, Section IV-A provides some preliminary results, which become the building blocks to prove Theorem 1. Then, the proof of Theorem 1 is given in Section IV-B.

A. Preliminary results

First, we show that the local dual function, $q_i(\lambda^i)$, for any $i \in \mathcal{N}$, is a Lipschitz smooth function.

Lemma 1: Let Assumptions 1-3 hold. The local dual function $q_i(\lambda^i)$ defined in (3) is Lipschitz smooth with Lipschitz constant $\frac{\|G^i\|^2}{\sigma_i}$. $\qquad \square$

Proof: Recall the definition of $q_i(\lambda^i)$ in (3) and let $u_i(\lambda^i) = \arg\min_{u_i \in \mathcal{U}_i}\left\{f_i(u_i) + \sum_{j \in \mathcal{M}_i} \langle G_j^{i\top}\lambda_j, u_i\rangle\right\}$ and $v_i(\mu^i) = \arg\min_{u_i \in \mathcal{U}_i}\left\{f_i(u_i) + \sum_{j \in \mathcal{M}_i} \langle G_j^{i\top}\mu_j, u_i\rangle\right\}$. Since $u_i(\lambda^i), v_i(\mu^i) \in \mathcal{U}_i$, the optimality conditions [21] of the preceding minimizations yield the following inequalities:

$$0 \leq \langle \nabla f_i(u_i(\lambda^i)) + G^{i\top}\lambda^i, v_i(\mu^i) - u_i(\lambda^i)\rangle, \qquad (15)$$

$$0 \leq \langle \nabla f_i(v_i(\mu^i)) + G^{i\top}\mu^i, u_i(\lambda^i) - v_i(\mu^i)\rangle. \qquad (16)$$

Combining (15) and (16) gives

$$\begin{aligned}
0 &\leq \langle \nabla f_i(u_i(\lambda^i)) - \nabla f_i(v_i(\mu^i)), v_i(\mu^i) - u_i(\lambda^i)\rangle \\
&\quad + \langle G^{i\top}(\lambda^i - \mu^i), v_i(\mu^i) - u_i(\lambda^i)\rangle \\
&\leq -\sigma_i \|v_i(\mu^i) - u_i(\lambda^i)\|^2 \\
&\quad + \langle \lambda^i - \mu^i, G^i(v_i(\mu^i) - u_i(\lambda^i))\rangle, \qquad (17)
\end{aligned}$$

where the second inequality is obtained since $f_i(\cdot)$ is strongly convex (c.f. Property 1). Furthermore, the strong convexity of $f_i(\cdot)$ also implies that $u_i(\lambda^i)$ is unique and $q_i(\lambda^i)$ is differentiable, with $\nabla q_i(\lambda^i) = G^i u_i(\lambda^i) - \tilde{g}^i$, where $\tilde{g}^i = \text{col}\{\tilde{g}_j^i, j \in \mathcal{M}_i\}$ and $\tilde{g}_j^i = 0_{m_j}$ if $j \neq i$ and $\tilde{g}_j^i = g_i$ otherwise. Thus, $\nabla q_i(\mu^i) - \nabla q_i(\lambda^i) = G^i(v_i(\mu^i) - u_i(\lambda^i))$. Using [8, Lemma 1.1] we obtain that

$$\frac{1}{\|G^i\|^2}\|\nabla q_i(\mu^i) - \nabla q_i(\lambda^i)\|^2 \leq \|v_i(\mu^i) - u_i(\lambda^i)\|^2. \quad (18)$$

By adding $\langle \lambda^i - \mu^i, \tilde{g}^i - \tilde{g}^i\rangle = 0$ to the right-hand side of (17), and then rearranging (17) as well as using (18) and the fact that $G^i v_i(\mu^i) - \tilde{g}^i = \nabla q_i(\mu^i)$ and $G^i u_i(\lambda^i) - \tilde{g}^i = \nabla q_i(\lambda^i)$, we obtain that

$$\begin{aligned}
&\frac{\sigma_i}{\|G^i\|^2}\|\nabla q_i(\mu^i) - \nabla q_i(\lambda^i)\|^2 \\
&\leq \langle \lambda^i - \mu^i, \nabla q_i(\mu^i) - \nabla q_i(\lambda^i)\rangle \\
&\leq \|\mu^i - \lambda^i\|\|\nabla q_i(\mu^i) - \nabla q_i(\lambda^i)\|,
\end{aligned}$$

where the second inequality is obtained using the Cauchy-Schwarz inequality. Thus, we have that

$$\|\nabla q_i(\mu^i) - \nabla q_i(\lambda^i)\| \leq \frac{\|G^i\|^2}{\sigma_i}\|\mu^i - \lambda^i\|,$$

showing that $q_i(\cdot)$ is Lipschitz smooth with Lipschitz constant $\frac{\|G^i\|^2}{\sigma_i}$ (c.f. Property 2). $\qquad \blacksquare$

Remark 1: The Lipschitz constant of $q_i(\cdot)$ can be computed locally by each agent $i \in \mathcal{N}$ since G^i and parameter σ_i are local information. $\qquad \square$

Lemma 2: Let Assumptions 1-3 hold. For any $\mu, \lambda \in \mathbb{R}^{\sum_{i \in \mathcal{N}} m_i}$, it holds that

$$q(\lambda) \geq q(\mu) + \langle \lambda - \mu, \nabla q(\mu)\rangle - \sum_{i \in \mathcal{N}} \frac{L_i}{2}\|\lambda_i - \mu_i\|^2, \quad (19)$$

where L_i, for each $i \in \mathcal{N}$, is defined in (12).

Proof: Since $q_i(\lambda_i)$ is concave and has a Lipschitz smooth gradient (Lemma 1), it follows from [22] that

$$q_i(\lambda^i) \geq q_i(\mu^i) + \langle \lambda^i - \mu^i, \nabla q_i(\mu^i)\rangle - \frac{\|G^i\|^2}{2\sigma_i}\|\lambda^i - \mu^i\|^2. \quad (20)$$

The desired inequality follows by summing (20) over $i \in \mathcal{N}$. $\qquad \blacksquare$

The Lipschitz smoothness property of the dual function (Lemma 2) is sufficient to show the inequality (22) stated in Lemma 3, which will become the key to prove Theorem 1. Note that Lemma 3 is similar to [7, Lemma 4.1] and [9, Lemma 5], although, differently from these references, the step-size η_i in (6) does not need to be the Lipschitz constant of the (dual) function.

Lemma 3: Let Assumptions 1-3 hold and the sequence $\{\theta(k), u_i(k), \lambda_i(k), \hat{\lambda}_i(k), \forall i \in \mathcal{N}\}$ be generated by Algorithm 1, with $\eta_i \in (0, 1/L_i]$, where L_i is defined by (12). Furthermore, let $\lambda^\star = \text{col}\{\lambda_i^\star, i \in \mathcal{N}\}$ be an optimal solution of the dual problem (4) and define $\omega_i(k)$ by

$$\omega_i(k) = \theta(k)\lambda_i(k) - (\theta(k) - 1)\lambda_i(k-1) - \lambda_i^\star, \quad (21)$$

for each $i \in \mathcal{N}$. Then, it holds that

$$\begin{aligned}
&\sum_{i \in \mathcal{N}} \frac{1}{2\eta_i}\left(\|\omega_i(k+1)\|^2 - \|\omega_i(k)\|^2\right) \leq \\
&\quad (\theta(k))^2(q(\lambda^\star) - q(\lambda(k))) \\
&\quad - (\theta(k+1))^2(q(\lambda^\star) - q(\lambda(k+1))).
\end{aligned} \qquad (22)$$

Proof: see Appendix A. $\qquad \blacksquare$

B. Proof of Theorem 1

Recall that α_i is the probability that the communication links between agent i and all its in-neighbors $j \in \mathcal{N}_i$ are active, i.e., $\alpha_i = \prod_{j \in \mathcal{N}_i} \beta_{\{i,j\}}$ and introduce the following function $V(k)$:

$$V(k) = \sum_{i \in \mathcal{N}} \frac{1}{2\alpha_i \eta_i} \|\omega_i(k)\|^2, \qquad (23)$$

where $\omega_i(k)$ is defined in (21).

To show the convergence, first we evaluate the sequence $\{\mathbb{E}(V(k))\}$. To this end, define $\mathcal{F}(k)$ as the filtration up to and including iteration k, i.e., $\mathcal{F}(k) = \{\mathcal{L}(\ell), \lambda(\ell), \xi(\ell), \ell = 0, 1, 2, \ldots, k\}$, where $\xi(k) = \mathrm{col}\{\xi^i(k), i \in \mathcal{N}\}$. Based on (9), $\lambda_i(k)$, for each $i \in \mathcal{N}$, is updated with the gradient ascent rule only when all the in-neighbors of agent i in $\mathcal{N}_i$ send new information to agent i. Otherwise, $\lambda_i(k) = \hat{\xi}_i^i(k)$. Therefore, if $\mathcal{N}_i \subseteq \mathcal{E}_i(k+1)$, $\omega_i(k+1)$ is computed using $\lambda_i(k+1)$ updated with the gradient ascent step. Otherwise, since $\lambda_i(k+1) = \hat{\xi}_i^i(k+1)$ (c.f. (9)), we have that

$$
\begin{aligned}
\omega_i(k+1) &= \theta(k+1)\hat{\xi}_i^i(k+1) - (\theta(k+1)-1)\lambda_i(k) - \lambda_i^\star \\
&= \theta(k+1)\lambda_i(k) + (\theta(k)-1)(\lambda_i(k)-\lambda_i(k-1)) \\
&\quad - (\theta(k+1)-1)\lambda_i(k) - \lambda_i^\star \\
&= \omega_i(k),
\end{aligned}
$$

where the second equality is obtained by using (11) and since $\lambda_i(k) = \xi_i^i(k)$, for any $k \geq 0$, due to (10) and a proper initialization in Algorithm 2.

Thus, we can see that $\omega(k+1)$ is updated with probability α_i and remains the same, i.e., $\omega_i(k+1) = \omega_i(k)$ with probability $1 - \alpha_i$. Based on this fact, we obtain, with probability 1, that

$$
\begin{aligned}
&\mathbb{E}\left(V(k+1) - V(k) | \mathcal{F}(k)\right) \\
&= \mathbb{E}\left(\sum_{i \in \mathcal{N}} \frac{1}{2\alpha_i \eta_i} \left(\|\omega_i(k+1)\|^2 - \|\omega_i(k)\|^2\right) \middle| \mathcal{F}(k)\right) \\
&= \sum_{i \in \mathcal{N}} \frac{1}{2\eta_i} \left(\frac{\alpha_i}{\alpha_i}\|\omega_i(k+1)\|^2 + \frac{1-\alpha_i}{\alpha_i}\|\omega_i(k)\|^2 \right. \\
&\qquad\qquad \left. - \frac{1}{\alpha_i}\|\omega_i(k)\|^2\right) \\
&= \sum_{i \in \mathcal{N}} \frac{1}{2\eta_i}\left(\|\omega_i(k+1)\|^2 - \|\omega_i(k)\|^2\right) \\
&\leq (\theta(k))^2(q(\lambda^\star) - q(\lambda(k))) \\
&\quad - (\theta(k+1))^2(q(\lambda^\star) - q(\lambda(k+1))), \qquad (24)
\end{aligned}
$$

where the inequality is obtained based on (22) in Lemma 3. Iterating (24), for $\ell = 1, 2, \ldots, k-1$, and taking the total
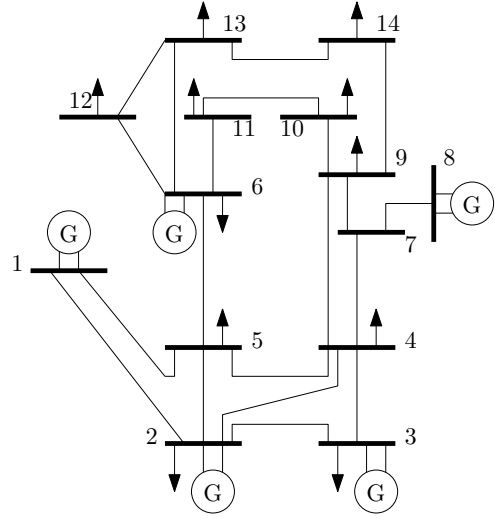

Fig. 2. The IEEE 14-bus network.

expectation, we have that

$$
\begin{aligned}
&\mathbb{E}\left(\sum_{\ell=1}^{k-1}(V(\ell+1) - V(\ell))\right) \\
&\leq \mathbb{E}\left(\sum_{\ell=1}^{k-1}(\theta(\ell))^2(q(\lambda^\star) - q(\lambda(\ell))) \right. \\
&\qquad\qquad \left. - (\theta(\ell+1))^2(q(\lambda^\star) - q(\lambda(\ell+1)))\right) \\
&\iff \mathbb{E}(V(k) - V(1)) \leq \theta(1)^2 \mathbb{E}\left(q(\lambda^\star) - q(\lambda(1))\right) \\
&\qquad\qquad - \mathbb{E}(\theta(k))^2(q(\lambda^\star) - q(\lambda(k)))). \qquad (25)
\end{aligned}
$$

Rearranging the inequality in (25) yields

$$
\begin{aligned}
&\mathbb{E}\left(\theta(k)^2(q(\lambda^\star) - q(\lambda(k)))\right) \\
&\leq \mathbb{E}(V(1) - V(k)) + \theta(1)^2 \mathbb{E}\left(q(\lambda^\star) - q(\lambda(1))\right) \\
&\leq \mathbb{E}(V(1) + q(\lambda^\star) - q(\lambda(1))), \qquad (26)
\end{aligned}
$$

where the second inequality is obtained since $\theta(1) = 1$ and by dropping $-\mathbb{E}(V(k))$ since it is non-positive for any $k \geq 1$. Finally, note that $\theta(k)$ is not random and it holds that $\theta(k) \geq \frac{k+1}{2}$ since $\theta(1) = 1$ and it is updated using the equation in step 6 of Algorithm 2 [7]. Using this fact and (26), the desired inequality (13) follows, where $C = 4\mathbb{E}(V(1) + q(\lambda^\star) - q(\lambda(1))) \geq 0$, since $\mathbb{E}(V(k)) \geq 0$, for any $k \geq 1$, and $q(\lambda^\star) = \max_\lambda q(\lambda)$, thus $\mathbb{E}(q(\lambda^\star) - q(\lambda(1))) \geq 0$. Upon obtaining (13), we can show the equality (14). Since C in (13) is non-negative, the term $\mathbb{E}(q(\lambda^\star) - q(\lambda(k)))$ converges to 0. Furthermore, using the Markov inequality, for any $\delta \in \mathbb{R}_{>0}$, we have that $\limsup_{k \to \infty} \mathbb{P}(q(\lambda^\star) - q(\lambda(k) \geq \delta) \leq \limsup_{k \to \infty} \frac{1}{\delta}\mathbb{E}(q(\lambda^\star) - q(\lambda(k)) = 0$, thus, $\lim_{k \to \infty} \mathbb{E}(q(\lambda^\star) - q(\lambda(k))) = 0$, almost surely. $\square$

V. NUMERICAL STUDY

We use the IEEE 14-bus benchmark case, which is shown in Figure 2, as the test case in this simulation study, where
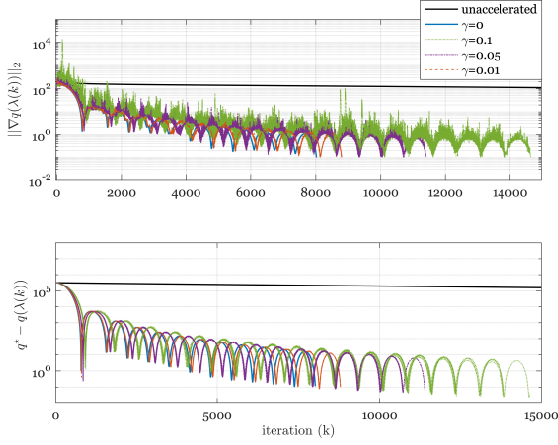
Fig. 3. Convergence of $\nabla q(\lambda(k))$ (top) and $q(\lambda(k)) - q^\star$ (bottom).

Fig. 4. The number of iterations performed for different values of γ. The blue boxes indicate the 25th-75th percentiles, the red lines indicate the median, and the + symbols indicate the outliers.

we solve an intra-day DC-OPF problem, with time horizon (h) of 6 hourly steps. We suppose that each bus is an agent in the network, though there are only five active agents, which have the capability of generating power, bounded by the capacity of the generators. Furthermore, we consider the DC-approximation of the power flow equations, as follows:

$$P_{i,t}^{\text{g}} - P_{i,t}^{\text{l}} = \sum_{j \in \mathcal{N}_i} B_{\{i,j\}}(\psi_{i,t} - \psi_{j,t}), \ \forall i \in \mathcal{N}, t = 1, \ldots, h, \tag{27}$$

where $P_{i,t}^{\text{g}} \in \mathbb{R}_{\geq 0}$ denotes the power generated at bus i at time step t, $P_{i,t}^{\text{l}} \in \mathbb{R}_{\geq 0}$ denotes the power demand assumed to be known for the whole time horizon, $B_{\{i,j\}}$ denotes the susceptance of line $\{i,j\}$, whereas ψ_i denotes the phase angle of bus i. The equalities in (27) become the coupling constraints of the network. In this problem, we compute the hourly set points of each generator for the whole time horizon. Additionally, we consider a strongly convex quadratic local cost.

We suppose that the communication links among the agents may fail with certain probability, denoted by $\gamma > 0$. This implies that the activation probability of each communication link is equal, i.e., $\beta_{\{i,j\}} = 1 - \gamma$, for each $i, j \in \mathcal{N}$, where $i \neq j$, and we perform 10 Monte-Carlo simulations for different values of γ. Moreover, we also compare Algorithm 2 with the unaccelerated version, where $\theta(k) = 1$ and $\gamma = 0$, for all $k \geq 1$. Figure 3 shows the convergence of the coupling constraint $\nabla q(\lambda(k))$ toward 0 and the dual value $q(\lambda(k))$ toward the optimal value $q^\star$. Additionally, Figure 4 shows the number of iterations required to meet the stopping criteria, which is the error of the equality constraint, i.e., $\|G_i^i u_i(k) + \sum_{j \in \mathcal{N}_i} G_i^j u_j(k) - g_i\| < \epsilon$, for a small $\epsilon \geq 0$. As expected, Algorithm 2 significantly outperforms the unaccelerated version, and the smaller γ, the faster convergence.

VI. CONCLUSION

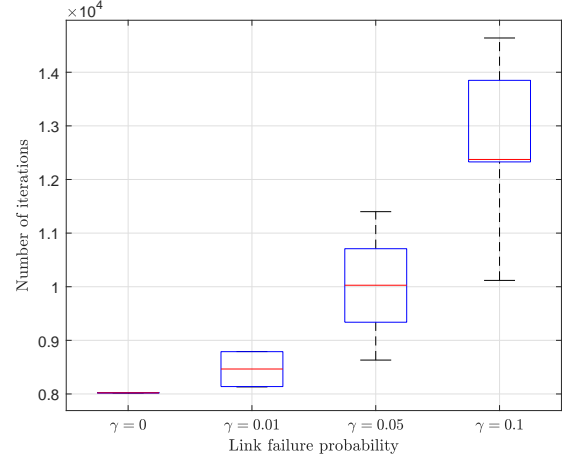In this paper, we propose a distributed algorithm for multi-agent optimization problem over stochastic networks. The algorithm is based on Nesterov's accelerated gradient method and we analytically show that the convergence rate of the expected dual value is $\mathcal{O}(1/k^2)$. We also show the performance of the algorithm in an intra-day optimal power flow simulation. As ongoing work, we are performing an analysis on the convergence of the primal variables. Moreover, we investigate methods to relax the assumptions considered to generalize the approach.

APPENDIX

A. Proof of Lemma 3

To show Lemma 3, we can follow the approach used on the proof of [7, Lemma 2.3]. Therefore, first we need the following intermediate result.

Lemma 4: Let $\psi(\mu, \xi)$ be a quadratic approximation model of $q(\mu)$, i.e.,

$$\psi(\mu, \xi) = q(\xi) + \langle \mu - \xi, \nabla q(\xi) \rangle - \sum_{i \in \mathcal{N}} \frac{1}{2\eta_i} \|\mu_i - \xi_i\|^2, \tag{28}$$

and $\lambda(\xi)$ be defined by $\lambda(\xi) = \arg\max_\mu \psi(\mu, \xi)$. Furthermore, let Assumptions 1-3 hold and $\eta_i \in (0, 1/L_i]$, where L_i is defined by (12). Then, for any $\mu \in \mathbb{R}^{\sum_{i \in \mathcal{N}} m_i}$,

$$q(\lambda(\xi)) - q(\mu) \geq \sum_{i \in \mathcal{N}} \frac{1}{\eta_i} \langle \xi_i - \mu_i, \lambda_i(\xi) - \xi_i \rangle$$
$$+ \sum_{i \in \mathcal{N}} \frac{1}{2\eta_i} \|\lambda_i(\xi) - \xi_i\|^2. \tag{29}$$

Proof: Since $\eta_i \in (0, 1/L_i]$, it follows from Lemma 2 that $q(\lambda(\xi)) \geq \psi(\lambda(\xi), \xi)$. Thus,

$$q(\lambda(\xi)) - q(\mu) \geq \psi(\lambda(\xi), \xi) - q(\mu).$$

Since $q(\cdot)$ is concave, we also have that

$$q(\mu) \leq q(\xi) + \langle \mu - \xi, \nabla q(\xi) \rangle.$$

The desired inequality (29) is obtained by combining the two preceding relations with the definition of $\psi(\lambda(\xi), \lambda)$ in (28) and $\lambda(\xi)$. ∎

Remark 2: The update $\lambda(k)$ in (6) follows $\lambda(k) = \arg\max_\mu \psi(\mu, \hat{\lambda}(k))$, which admits a unique solution. □
Next, [9, Lemma 4] shows that $\omega_i(k+1) = \omega_i(k) + \theta(k+1)\left(\lambda_i(k+1) - \hat{\lambda}_i(k+1)\right)$. Based on this relation, we obtain that

$$\|\omega_i(k+1)\|^2 - \|\omega_i(k)\|^2$$
$$= \|\omega_i(k) + \theta(k+1)(\lambda_i(k+1) - \hat{\lambda}_i(k+1))\|^2 - \|\omega_i(k)\|^2$$
$$= 2\theta(k+1)(\theta(k+1) - 1)\cdot$$
$$\cdot \langle \lambda_i(k+1) - \hat{\lambda}_i(k+1), \hat{\lambda}_i(k+1) - \lambda_i(k)\rangle +$$
$$+ (\theta(k+1)^2 - \theta(k+1))\|\lambda_i(k+1) - \hat{\lambda}_i(k+1)\|^2 +$$
$$+ \theta(k+1)\|\lambda_i(k+1) - \hat{\lambda}_i(k+1)\|^2 +$$
$$+ 2\theta(k+1)\langle \lambda_i(k+1) - \hat{\lambda}_i(k+1), \hat{\lambda}_i(k+1) - \lambda_i^\star\rangle,$$

where the second equality is obtained by performing some algebraic manipulations using (21) and (7). Multiplying by $\frac{1}{2\eta_i}$ and summing over $i \in \mathcal{N}$ the above equality, we obtain that

$$\sum_{i\in\mathcal{N}} \frac{1}{2\eta_i}\left(\|\omega_i(k+1)\|^2 - \|\omega_i(k)\|^2\right)$$
$$= (\theta(k+1)^2 - \theta(k+1))\cdot$$
$$\sum_{i\in\mathcal{N}}\left(\frac{1}{\eta_i}\langle \lambda_i(k+1) - \hat{\lambda}_i(k+1), \hat{\lambda}_i(k+1) - \lambda_i(k)\rangle\right.$$
$$\left. + \frac{1}{2\eta_i}\|\lambda_i(k+1) - \hat{\lambda}_i(k+1)\|^2\right)$$
$$+ \theta(k+1)\sum_{i\in\mathcal{N}}\left(\frac{1}{2\eta_i}\|\lambda_i(k+1) - \hat{\lambda}_i(k+1)\|^2\right.$$
$$\left. + \frac{1}{\eta_i}\langle \lambda_i(k+1) - \hat{\lambda}_i(k+1), \hat{\lambda}_i(k+1) - \lambda_i^\star\rangle\right).$$

By applying the inequality (29) twice to substitute each term inside the two summations, we obtain the desired inequality, as follows:

$$\sum_{i\in\mathcal{N}} \frac{1}{2\eta_i}\left(\|\omega_i(k+1)\|^2 - \|\omega_i(k)\|^2\right)$$
$$\leq (\theta(k+1)^2 - \theta(k+1))(q(\lambda(k+1)) - q(\lambda(k))$$
$$+ \theta(k+1)(q(\lambda(k+1)) - q(\lambda^\star))$$
$$= \theta(k+1)^2 q(\lambda(k+1)) - (\theta(k+1)^2 - \theta(k+1))q(\lambda(k))$$
$$- \theta(k+1)q(\lambda^\star)$$
$$= \theta(k+1)^2 q(\lambda(k+1)) - \theta(k)^2 q(\lambda(k))$$
$$+ (\theta(k)^2 - \theta(k+1)^2)q(\lambda^\star)$$
$$= \theta(k)^2(q(\lambda^\star) - q(\lambda(k)))$$
$$- \theta(k+1)^2(q(\lambda^\star) - q(\lambda(k+1))),$$

where the second equality is obtained based on step 4 of Algorithm 1, where $\theta(k+1)^2 - \theta(k+1) - \theta(k)^2 = 0$. □

REFERENCES

[1] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A survey of distributed optimization and control algorithms for electric power systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.

[2] P. Yi, Y. Hong, and F. Liu, "Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems," *Automatica*, vol. 74, pp. 259–269, 2016.

[3] J. M. Grosso, C. Ocampo-Martinez, and V. Puig, "A distributed predictive control approach for periodic flow-based networks: application to drinking water systems," *International Journal of Systems Science*, vol. 48, no. 14, pp. 3106–3117, 2017.

[4] L. Xiao and S. Boyd, "Optimal scaling of a gradient method for distributed resource allocation," *Journal of Optimization Theory and Applications*, vol. 129, pp. 469–488, 2006.

[5] D. P. Bertsekas, "Extended monotropic programming and duality," *Journal of Optimization Theory and Applications*, vol. 139, pp. 209–225, 2008.

[6] Y. Nesterov, "A method for solving the convex programming problem with convergence rate $O(1/k^2)$," *Dokl. Akad. Nauk SSSR*, vol. 27, p. 543–547, 1983, translated as Sov. Math. Dokl.

[7] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[8] A. Beck, A. Nedić, A. Ozdaglar, and M. Teboulle, "An $o(1/k)$ gradient method for network resource allocation problems," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 64–73, 2014.

[9] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk, "Fast alternating direction optimization methods," *SIAM Journal on Imaging Sciences*, vol. 7, no. 3, pp. 1588–1623, 2014.

[10] X. Zhou, C. Li, T. Huang, and M. Xiao, "Fast gradient-based distributed optimisation approach for model predictive control and application in four-tank benchmark," *IET Control Theory Applications*, vol. 9, no. 10, pp. 1579–1586, 2015.

[11] E. Wei and A. Ozdaglar, "On the O(1/k) convergence of asynchronous distributed alternating direction method of multipliers," pp. 1–30, 2013, arXiv:1307.8254.

[12] T. Chang, M. Hong, W. Liao, and X. Wang, "Asynchronous distributed ADMM for large-scale optimization—Part I: algorithm and convergence analysis," *IEEE Transactions on Signal Processing*, vol. 64, no. 12, pp. 3118–3130, 2016.

[13] M. Hong and T. Chang, "Stochastic proximal gradient consensus over random networks," *IEEE Transactions on Signal Processing*, vol. 65, no. 11, pp. 2933–2948, 2017.

[14] W. Ananduta, A. Nedić, and C. Ocampo-Martinez, "Distributed augmented Lagrangian method for link-based resource sharing problems of multi-agent systems," *IEEE Transactions on Automatic Control*, submitted.

[15] D. Jakovetić, J. M. F. Xavier, and J. M. F. Moura, "Convergence rates of distributed nesterov-like gradient methods on random networks," *IEEE Transactions on Signal Processing*, vol. 62, no. 4, pp. 868–882, 2014.

[16] O. Fercoq and P. Richtárik, "Accelerated, parallel, and proximal coordinate descent," *SIAM Journal on Optimization*, vol. 25, no. 4, pp. 1997–2023, 2015.

[17] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 601–615, 2015.

[18] C. A. Uribe, S. Lee, A. Gasnikov, and A. Nedić, "A dual approach for optimal algorithms in distributed optimization over networks," *Optimization Methods and Software*, pp. 1–40, 2020.

[19] G. Scutari and Y. Sun, "Distributed nonconvex constrained optimization over time-varying digraphs," *Mathematical Programming*, vol. 176, pp. 497–544, 2019.

[20] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1995.

[21] A. Nedić, *Lecture Notes Optimization I*. Hamilton Institute, 2008.

[22] X. Zhou, "On the fenchel duality between strong convexity and lipschitz continuous gradient," pp. 1–6, 2018, arXiv:1803.06573.