Distributed Dual Gradient Tracking for Resource Allocation in Unbalanced Networks

Jiaqi Zhang , Keyou You , *Senior Member, IEEE*, and Kai Cai , *Senior Member, IEEE*

Abstract—This paper proposes a distributed dual gradient tracking algorithm (DDGT) to solve resource allocation problems over an unbalanced network, where each node in the network holds a private cost function and computes the optimal resource by interacting only with its neighboring nodes. Our key idea is the novel use of the distributed push-pull gradient algorithm (PPG) to solve the dual problem of the resource allocation problem. To study the convergence of the DDGT, we first establish the sublinear convergence rate of PPG for *non-convex* objective functions, which advances the existing results on PPG as they require the strong-convexity of objective functions. Then we show that the DDGT converges linearly for strongly convex and Lipschitz smooth cost functions, and sublinearly without the Lipschitz smoothness. Finally, experimental results suggest that DDGT outperforms existing algorithms.

Index Terms—Distributed resource allocation, unbalanced graphs, dual problem, distributed optimization, push-pull gradient.

I. INTRODUCTION

DISTRIBUTED resource allocation problems (DRAPs) are concerned with optimally allocating resources to multiple nodes that are connected via a directed peer-to-peer network. Each node is associated with a local private objective function to measure the cost of its allocated resource, and the global goal is to jointly minimize the total cost. The key feature of the DRAPs is that each node computes its optimal amount of resources by interacting *only* with its neighboring nodes in the network. A typical application is the economic dispatch problem, where the local cost function is often quadratic [1]. See [2]–[5] for other applications.

A. Literature Review

Existing works on DRAPs can be categorized depending on whether the underlying network is *balanced* or not. A balanced network means that the "amount" of information *to* any node is equal to that *from* this node, which is crucial to the algorithm design. Most of early works on DRAPs focus on balanced networks and the recent interest is shifted to the unbalanced case.

The central-free algorithm (CFA) in [2] is the first documented result on DRAPs for balanced networks, where at each iteration every node updates its decision variables using the weighted error between the gradient of its local objective function and those of its neighbors, and it can be accelerated by designing an optimal weighting matrix [3]. It is proved that the CFA achieves a linear convergence rate for strongly convex and Lipschitz smooth cost functions. For time-varying networks, the CFA is shown to converge sublinearly in the absence of strong convexity [4]. This rate is further improved in [6] by optimizing its dependence on the number of nodes. In addition, there are also several ADMM-based methods that only work for balanced networks [7]–[9]. By exploiting the mirror relationship between the distributed optimization and distributed resource allocation, several accelerated distributed algorithms are proposed in [10], [11]. Moreover, [12] and [13] study continuous-time algorithms for DRAPs by using the machinery of control theory.

For unbalanced networks, the algorithm design for DRAPs is much more complicated, which has been widely acknowledged in the distributed optimization literature [14], [15]. In this case, a consensus based algorithm that adopts the celebrated surplus idea [15] is proposed in [1] and [16]. However, their convergence results are only for quadratic cost functions where the analyses rely on the linear system theory. The extension to general convex functions is performed in [17] by adopting the nonnegative surplus method, at the expense of a slower convergence rate. The ADMM-based algorithms are developed in [18], [19], and algorithms that aim to handle communication delay in time-varying networks and perform event-triggered updates are respectively studied in [20] and [21]. We note that all the above-mentioned works [1], [16]–[21] do not provide explicit convergence rates for their algorithms. In contrast, the algorithm proposed in this work is proved to achieve a linear convergence rate for strongly convex and Lipschitz smooth cost functions, and has a sublinear convergence rate without the Lipschitz smoothness.

There are several recent works with convergence rate analyses for their algorithms over unbalanced networks. Most of them leverage the dual relationship between DRAPs and distributed optimization problems. For example, the algorithms in [22] and [23] use stochastic gradients and diminishing stepsize to

solve the dual problem of DRAPs, and thus their convergence rates are limited to an order of $O(\ln(k)/\sqrt{k})$ for Lipschitz smooth cost functions. [23] also shows a rate of $O(\ln(k)/k)$ if the cost function is strongly convex. An algorithm with linear convergence rate is recently proposed in [24] for strongly convex and Lipschitz smooth cost functions. However, its convergence rate is unclear if either the strongly convexity or the Lipschitz smoothness is removed. In [9], a push-sum-based algorithm is proposed by incorporating the alternating direction method of multipliers (ADMM). Although it can handle time-varying networks, the convergence rate is $O(1/k)$ even for strongly convex and Lipschitz smooth functions.

B. Our Contributions

In this work, we propose a distributed dual gradient tracking algorithm (DDGT) to solve DRAPs over unbalanced networks. The DDGT exploits the duality of DRAPs and distributed optimization problems, and takes advantage of the distributed push-pull gradient algorithm (PPG) [25], which is also called $\mathcal{AB}$ algorithm in [26]. If the cost function is strongly convex and Lipschitz smooth, we show that the DDGT converges at a linear rate $O(\lambda^k), \lambda \in (0, 1)$. If the Lipschitz smoothness is not satisfied, we show the convergence of the DDGT and establish an convergence rate $O(1/k)$. To our best knowledge, these convergence results are only reported for *undirected* or *balanced* networks in [10]. Although a distributed algorithm for directed networks is also proposed in [10], there is no convergence analysis. The advantages of the DDGT over existing algorithms are also validated by numerical experiments.

To characterize the sublinear convergence of the DDGT, we first show that PPG converges sublinearly to a stationary point even for *non-convex* objective functions. Clearly, this advances existing works [25]–[27] as their convergence results are only for strongly-convex objective functions. In fact, the convergence proofs for PPG in [25]–[27] require constructing a complicated 3-dimensional matrix and then derive the linear convergence rate $O(\lambda^k)$ where $\lambda \in (0, 1)$ is the spectral radius of this matrix. This approach is no longer applicable since a linear convergence rate is usually not attainable for general non-convex functions [28] and hence the spectral radius of such a matrix cannot be strictly less than one.

C. Paper Organization and Notations

The rest of this paper is organized as follows. In Section II, we formulate the constrained DRAPs with some standard assumptions. Section III firstly derives the dual problem of DRAPs which is amenable to distributed optimization, and then introduces the PPG. The DDGT is then obtained by applying PPG to the dual problem and improving the initialization. In Section IV, the convergence result of the DDGT is derived by establishing the convergence of PPG for non-convex objective functions. Section V performs numerical experiments to validate the effectiveness of the DDGT. Finally, we draw conclusive remarks in Section VI.

We use a lowercase x, bold letter $\mathbf{x}$ and uppercase X to denote a scalar, vector, and matrix, respectively. $\mathbf{x}^{\mathsf{T}}$ denotes the transpose of the vector $\mathbf{x}$. $[X]_{ij}$ denotes the element in the i-th row and j-th column of the matrix X. For vectors we use $\|\cdot\|$ to denote the l_2-norm. For matrices we use $\|\cdot\|$ and $\|\cdot\|_F$ to denote respectively the spectral norm and the Frobenius norm. $|\mathcal{X}|$ denotes the cardinality of set $\mathcal{X}$. $\mathbb{R}^n$ denotes the set of n-dimensional real vectors. $\mathbf{1}$ denotes the vector with all ones, the dimension of which depends on the context. $\nabla f(\mathbf{x})$ denotes the gradient of a differentiable function f at $\mathbf{x}$. We say a nonnegative matrix X is row-stochastic if $X\mathbf{1} = \mathbf{1}$, and column-stochastic if X^{T} is row-stochastic. $O(\cdot)$ denotes the big-O notation.

II. PROBLEM FORMULATION

Consider the distributed resource allocation problems (DRAPs) with n nodes, where each node i has a local private cost function $F_i : \mathbb{R}^m \to \mathbb{R}$. The goal is to solve the following optimization problem in a distributed manner:

$$\underset{\mathbf{w}_1,\dots,\mathbf{w}_n \in \mathbb{R}^m}{\text{minimize}} \quad \sum_{i=1}^n F_i(\mathbf{w}_i)$$

$$\text{subject to} \quad \mathbf{w}_i \in \mathcal{W}_i, \ \sum_{i=1}^n \mathbf{w}_i = \sum_{i=1}^n \mathbf{d}_i \tag{1}$$

where $\mathbf{w}_i \in \mathbb{R}^m$ is the local decision vector of node i, representing the resources allocated to i. $\mathcal{W}_i$ is a local convex and closed constraint set. $\mathbf{d}_i$ denotes the resource demand of node i. Both $\mathcal{W}_i$ and $\mathbf{d}_i$ are only known to node i. Let $\mathbf{d} \triangleq \sum_{i=1}^n \mathbf{d}_i$, then $\sum_{i=1}^n \mathbf{w}_i = \mathbf{d}$ represents the constraint on total available resources, showing the coupling among nodes.

Remark 1: Problem (1) covers many forms of DRAPs considered in the literature. For example, the standard local constraint $\mathcal{W}_i = [\underline{w}_i, \overline{w}_i]$ for some constants $\underline{w}_i$ and $\overline{w}_i$ is a one-dimensional special case of (1), see e.g. [1], [16], [17], [20], [24]. Moreover, the coupling constraint can be given in a weighted form $\sum_{i=1}^n A_i \mathbf{w}_i = \mathbf{d}$, which can be transformed into (1) by defining a new variable $\mathbf{w}_i' = A_i \mathbf{w}_i$ and a local constraint set $\mathcal{W}_i' = \{A_i \mathbf{w}_i | \mathbf{w}_i \in \mathcal{W}_i\}$. In addition, many works only consider quadratic cost functions [1], [16].

Solving (1) in a distributed manner means that each node can only communicate and exchange information with a subset of nodes via a communication network, which is modeled by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Here $\mathcal{V} = \{1, \dots, n\}$ denotes the set of nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges, and $(i, j) \in \mathcal{E}$ if node i can send information to node j. Note that $(i, j) \in \mathcal{E}$ does not necessarily imply that $(j, i) \in \mathcal{E}$. Define $\mathcal{N}_i^{\text{in}} = \{j | (j, i) \in \mathcal{E}\} \cup \{i\}$ and $\mathcal{N}_i^{\text{out}} = \{j | (i, j) \in \mathcal{E}\} \cup \{i\}$ as the set of in-neighbors and out-neighbors of node i, respectively. That is, node i can only receive messages from its in-neighbors and send messages to its out-neighbors. Let $a_{ij} > 0$ be the weight associated to edge $(j, i) \in \mathcal{E}$. $\mathcal{G}$ is *balanced* if $\sum_{j \in \mathcal{N}_i^{\text{in}}} a_{ij} = \sum_{j \in \mathcal{N}_i^{\text{out}}} a_{ji}$ for all $i \in \mathcal{V}$. Note that balancedness is a relatively strong condition, since it can be difficult or even impossible to find weights satisfying it for a general directed graph [29].

The following assumptions are made throughout the paper.

Assumption 1 (Strong convexity and Slater's condition):

1) The local cost function F_i is μ-strongly convex for all $i \in \mathcal{V}$, i.e., for any $\mathbf{w}, \mathbf{w}' \in \mathbb{R}^m$ and $\theta \in [0,1]$,

$$F_i(\theta\mathbf{w} + (1-\theta)\mathbf{w}')$$
$$\leq \theta F_i(\mathbf{w}) + (1-\theta)F_i(\mathbf{w}') - \frac{\mu}{2}\theta(1-\theta)\|\mathbf{w} - \mathbf{w}'\|^2.$$

2) The constraint $\sum_{i=1}^n \mathbf{w}_i = \mathbf{d}$ is satisfied for some point in the relative interior of the Cartesian product $\mathcal{W} := \mathcal{W}_1 \times \cdots \times \mathcal{W}_n$.

Assumption 2 (Strongly connected network): $\mathcal{G}$ is strongly connected, i.e., there exists a directed path from any node i to any node j.

Assumption 1 is common in the literature. Note that we do not assume the differentiability of F_i. Under Assumption 1, the optimal point of (1) is unique. Let $F^\star$ and $\mathbf{w}_i^\star, i \in \mathcal{V}$ denote respectively its optimal value and optimal point, i.e., $F^\star = \sum_{i=1}^n F_i(\mathbf{w}_i^\star)$. Assumption 2 is also common and necessary for the information mixing over a network.

III. THE DISTRIBUTED DUAL GRADIENT TRACKING ALGORITHM

This section introduces our distributed dual gradient tracking algorithm (DDGT) to solve (1) over an unbalanced network. We start with the dual problem of (1) and transform it into a form for distributed optimization. Then, the DDGT is obtained by using the push-pull gradient method (PPG [25], [26]) on the dual problem, which is an efficient distributed optimization algorithm over *unbalanced* networks.

A. The Dual Problem of (1) and PPG

Define the Lagrange function of (1) as

$$L(W, \mathbf{x}) = \sum_{i=1}^n F_i(\mathbf{w}_i) + \mathbf{x}^\mathsf{T}\left(\sum_{i=1}^n \mathbf{w}_i - \mathbf{d}\right) \qquad (2)$$

where $W = [\mathbf{w}_1, \ldots, \mathbf{w}_n] \in \mathbb{R}^{m \times n}$ and $\mathbf{x}$ is the Lagrange multiplier. Then, the dual problem of (1) is given by

$$\underset{\mathbf{x} \in \mathbb{R}^m}{\text{maximize}} \; \underset{W \in \mathcal{W}}{\inf} \; L(W, \mathbf{x}). \qquad (3)$$

Under Assumption 1, the strong duality holds [30], [31, Exercise 5.2.2]. The objective function in (3) is written as

$$\underset{W \in \mathcal{W}}{\inf} L(W, \mathbf{x}) = \underset{W \in \mathcal{W}}{\inf} \sum_{i=1}^n \left(F_i(\mathbf{w}_i) + \mathbf{x}^\mathsf{T}\mathbf{w}_i\right) - \mathbf{x}^\mathsf{T}\mathbf{d}$$

$$= \sum_{i=1}^n \underset{\mathbf{w}_i \in \mathcal{W}_i}{\inf} \left\{F_i(\mathbf{w}_i) + \mathbf{x}^\mathsf{T}\mathbf{w}_i\right\} - \mathbf{x}^\mathsf{T}\mathbf{d}$$

$$= \sum_{i=1}^n -F_i^*(-\mathbf{x}) - \mathbf{x}^\mathsf{T}\mathbf{d}$$

where

$$F_i^*(\mathbf{x}) \triangleq \underset{\mathbf{w}_i \in \mathcal{W}_i}{\sup} \left\{\mathbf{w}_i^\mathsf{T}\mathbf{x} - F_i(\mathbf{w}_i)\right\} \qquad (4)$$

is the convex conjugate function corresponding to the pair $(F_i, \mathcal{W}_i)$ [31, Section 5.4]. Thus, the dual problem (3) can be rewritten as a convex optimization problem

$$\underset{\mathbf{x} \in \mathbb{R}^m}{\text{minimize}} \; f(\mathbf{x}) \triangleq \sum_{i=1}^n f_i(\mathbf{x}), \; f_i(\mathbf{x}) \triangleq F_i^*(-\mathbf{x}) + \mathbf{x}^\mathsf{T}\mathbf{d}_i \quad (5)$$

or equivalently,

$$\underset{\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^m}{\text{minimize}} \; \sum_{i=1}^n f_i(\mathbf{x}_i)$$

$$\text{subject to} \quad \mathbf{x}_1 = \cdots = \mathbf{x}_n. \qquad (6)$$

Recall that strong duality holds, and therefore problem (6) is equivalent to problem (1) in the sense that the optimal value of (6) is $f^\star = -F^\star$ and the optimal point $\mathbf{x}_1^\star = \cdots = \mathbf{x}_n^\star = \mathbf{x}^\star$ of (6) satisfies $F_i(\mathbf{w}_i^\star) + F_i^*(-\mathbf{x}^\star) = -(\mathbf{w}_i^\star)^\mathsf{T}\mathbf{x}^\star$. Hence, we can simply focus on solving the dual problem (6).

The strong convexity of F_i implies that F_i^* is differentiable with Lipschitz continuous gradients [30], and the supremum in (4) is attainable. By Danskin's theorem [31], the gradient of F_i^* is given by $\nabla F_i^*(\mathbf{x}) = \operatorname{argmax}_{\mathbf{w} \in \mathcal{W}_i}\{\mathbf{x}^\mathsf{T}\mathbf{w} - F_i(\mathbf{w})\}$. Thus, it follows from (5) that

$$\nabla f_i(\mathbf{x}) = -\nabla F_i^*(-\mathbf{x}) + \mathbf{d}_i$$

$$= -\underset{\mathbf{w} \in \mathcal{W}_i}{\operatorname{argmin}}\{\mathbf{x}^\mathsf{T}\mathbf{w} + F_i(\mathbf{w})\} + \mathbf{d}_i. \qquad (7)$$

The dual form (6) allows us to take advantage of recent advances in distributed optimization to solve DRAPs over *unbalanced* networks. For example, distributed algorithms are proposed in [32, gradient-push], [33, Push-DIGing], [34, ExtraPush], [35, DEXTRA], [36] to solve (6) over general directed and unbalanced graphs. Asynchronous algorithms are also studied in [37]–[40]. In particular, [25] and [26] propose PPG algorithm (or called $\mathcal{AB}$ in [26]) by using the idea of gradient tracking, which achieves a linear convergence rate if the objective function f_i is strongly convex and Lipschitz smooth for all i. Moreover, PPG has an empirically faster convergence speed than its competitors (e.g. [33]), and its linear update rule is an advantage for implementation. The compact form of PPG is given as

$$\mathbf{x}_{k+1}^{(i)} = \sum_{j \in \mathcal{N}_i^{\text{in}}} a_{ij}\left(\mathbf{x}_k^{(j)} - \alpha\mathbf{y}_k^{(j)}\right)$$

$$\mathbf{y}_{k+1}^{(i)} = \sum_{j \in \mathcal{N}_i^{\text{in}}} b_{ij}\mathbf{y}_k^{(j)} + \nabla f_i\left(\mathbf{x}_{k+1}^{(i)}\right) - \nabla f_i\left(\mathbf{x}_k^{(i)}\right) \quad (8)$$

where $a_{ij} > 0$ for any $j \in \mathcal{N}_i^{\text{in}}$ and $\sum_{j \in \mathcal{N}_i^{\text{in}}} a_{ij} = 1$, $b_{ij} > 0$ for any $i \in \mathcal{N}_j^{\text{out}}$ and $\sum_{i \in \mathcal{N}_j^{\text{out}}} b_{ij} = 1$, α is a positive stepsize, and $\mathbf{x}_0^{(i)}$ and $\mathbf{y}_0^{(i)}$ are initialized such that $\mathbf{y}_0^{(i)} = \nabla f_i(\mathbf{x}_0^{(i)}), \forall i \in \mathcal{V}$. Intuitively, the update for $\mathbf{y}_k^{(i)}$ aims to asymptotically track the global gradient $\nabla f(\bar{\mathbf{x}}_k)$ and the update for $\mathbf{x}_k^{(i)}$ enforces it to converge to $\bar{\mathbf{x}}_k$ while performing an inexact gradient descent step, where $\bar{\mathbf{x}}_k = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_k^{(i)}$ is the mean of nodes' states. We refer interested readers to [25], [26] for more discussions on PPG.

Algorithm 1: The Distributed Dual Gradient Tracking Algorithm (DDGT) — From the View of Node i.

- **Initialization:** Let $\overline{\mathbf{w}}_0^{(i)} = 0$, $\mathbf{w}_0^{(i)} = 0$, $\mathbf{s}_0^{(i)} = \mathbf{d}_i$.[a]
- **For** $k = 0, 1, \ldots, K$, **repeat**
  1: Receive $\widetilde{\mathbf{w}}_k^{(j)} := \overline{\mathbf{w}}_k^{(j)} + \alpha \mathbf{s}_k^{(j)}$ and $\widetilde{\mathbf{s}}_k^{(ji)} := b_{ij} \mathbf{s}_k^{(j)}$ from its in-neighbor j.
  2: Compute $\overline{\mathbf{w}}_{k+1}^{(i)}$, $\mathbf{w}_{k+1}^{(i)}$ and $\mathbf{s}_{k+1}^{(i)}$ as (9).
  3: Broadcast $\widetilde{\mathbf{w}}_{k+1}^{(i)}$ and $\widetilde{\mathbf{s}}_{k+1}^{(i)}$ to each of out-neighbors.
- **Return** $\mathbf{w}_K^{(i)}$.

---

[a]If only the total resource demand $\mathbf{d}$ is known to all nodes, then we can simply set $\mathbf{s}_0^{(i)} = \frac{1}{n}\mathbf{d}$, which can be done in a distributed manner [17].

B. The DDGT Algorithm

We are ready to present the DDGT algorithm. Plugging the gradient (7) into (8) and noticing that the $\mathbf{d}_i$ term is cancelled in $\nabla f_i(\mathbf{x}_{k+1}^{(i)}) - \nabla f_i(\mathbf{x}_k^{(i)})$, we have

$$\overline{\mathbf{w}}_{k+1}^{(i)} = \sum_{j \in \mathcal{N}_i^{\text{in}}} a_{ij} \left(\overline{\mathbf{w}}_k^{(j)} + \alpha \mathbf{s}_k^{(j)} \right), \tag{9a}$$

$$\mathbf{w}_{k+1}^{(i)} = \underset{\mathbf{w} \in \mathcal{W}_i}{\arg\min} \left\{ F_i(\mathbf{w}) - \mathbf{w}^{\mathsf{T}} \overline{\mathbf{w}}_{k+1}^{(i)} \right\}, \tag{9b}$$

$$\mathbf{s}_{k+1}^{(i)} = \sum_{j \in \mathcal{N}_i^{\text{in}}} b_{ij} \mathbf{s}_k^{(j)} - \left(\mathbf{w}_{k+1}^{(i)} - \mathbf{w}_k^{(i)} \right). \tag{9c}$$

where notations have been changed to keep consistency with the primal problem (1), i.e., $\mathbf{x}_k^{(i)} = -\overline{\mathbf{w}}_k^{(i)}$ and $\mathbf{y}_k^{(i)} = \mathbf{s}_k^{(i)}$.

The DDGT is summarized in Algorithm 1 and we now elaborate on it. After initialization, each node i iteratively updates three vectors $\overline{\mathbf{w}}_k^{(i)}$, $\mathbf{w}_k^{(i)}$ and $\mathbf{s}_k^{(i)}$. In particular, at each iteration node i receives $\widetilde{\mathbf{w}}_k^{(j)} := \overline{\mathbf{w}}_k^{(j)} + \alpha \mathbf{s}_k^{(j)}$ and $\widetilde{\mathbf{s}}_k^{(ji)} := b_{ij} \mathbf{s}_k^{(j)}$ from each of its in-neighbors j, and updates $\overline{\mathbf{w}}_{k+1}^{(i)}$ according to (9a), where a_{ij} is positive for any $j \in \mathcal{N}_i^{\text{in}}$ such that $\sum_{j \in \mathcal{N}_i^{\text{in}}} a_{ij} = 1$ as with (8), and α is a positive stepsize. The update of $\mathbf{s}_k^{(i)}$ in (9c) is similar, where $b_{ij} > 0$ for any $i \in \mathcal{N}_j^{\text{out}}$ and $\sum_{i \in \mathcal{N}_j^{\text{out}}} b_{ij} = 1$. This process repeats until terminated. We set $a_{ij} = b_{ij} = 0$ for any $(j, i) \notin \mathcal{E}$ for convenience. Define two matrices $[A]_{ij} = a_{ij}$ and $[B]_{ij} = b_{ij}$, then A is a row-stochastic matrix and B is a column-stochastic matrix. Clearly, the directed network associated with A and B can be unbalanced.

Remark 2: In practice, one can simply set $a_{ij} = |\mathcal{N}_i^{\text{in}}|^{-1}$ and $b_{ij} = |\mathcal{N}_j^{\text{out}}|^{-1}$, and then all conditions are satisfied. Note that this setting requires each node to know the number of its in-neighbors and out-neighbors, which is common in the literature of distributed optimization over directed networks [32]–[35].

Notably, the initialization for DDGT exploits the structure of the DRAPs and improves that of PPG. By PPG, $\mathbf{w}_0^{(i)}$ and $\mathbf{s}_0^{(i)}$ should be exactly set as $\mathbf{w}_0^{(i)} = \widetilde{\mathbf{w}}_i^{\star}$ and $\mathbf{s}_0^{(i)} = \mathbf{d}_i - \widetilde{\mathbf{w}}_i^{\star}$, where $\widetilde{\mathbf{w}}_i^{\star} = \arg\min_{\mathbf{w} \in \mathcal{W}_i} F_i(\mathbf{w})$ is a local minimizer. In DDGT, the computation of $\widetilde{\mathbf{w}}_i^{\star}$ is actually not necessary since the update without $\widetilde{\mathbf{w}}_i^{\star}$ in $\mathbf{w}_0^{(i)}$ and $\mathbf{s}_0^{(i)}$ and the update with it become

equivalent after the first iteration due to the special form of $\nabla f_i(\mathbf{x})$. Clearly, the former is simpler and is adopted in DDGT.

The update of $\mathbf{w}_k^{(i)}$ in (9b) requires finding an optimal point of an auxiliary *local* optimization problem, which can be obtained by standard algorithms, e.g., projected (sub)gradient method or Newton's method, and can even be given in an explicit form for some special cases. Note that solving sub-problems per iteration is common in many duality-based optimization algorithms, including the dual ascent method and proximal method [41].

Remark 3: Consider two special cases. The first one is that the local constraint set $\mathcal{W}_i = \mathbb{R}^m$ and F_i is differentiable as in [4]. Then, (9b) becomes

$$\mathbf{w}_{k+1}^{(i)} = \nabla^{-1} F_i(\overline{\mathbf{w}}_{k+1}^{(i)}) \tag{9b$'$}$$

where $\nabla^{-1} F_i$ denotes the inverse function of ∇F_i, i.e., $\nabla^{-1} F_i(\nabla F_i(\mathbf{x})) = \mathbf{x}$ for any $\mathbf{x} \in \mathbb{R}^m$.

The second case is that the decision variable is a scalar, $\mathcal{W}_i$ is an interval $[\underline{w}_i, \overline{w}_i]$, and F_i is differentiable as in [1], [17], [20]. Then, (9b) becomes

$$\mathbf{w}_{k+1}^{(i)} = \begin{cases} \overline{w}_i, & \text{if } \nabla^{-1} F(\overline{\mathbf{w}}_{k+1}^{(i)}) > \overline{w}_i \\ \underline{w}_i, & \text{if } \nabla^{-1} F(\overline{\mathbf{w}}_{k+1}^{(i)}) < \underline{w}_i \\ \nabla^{-1} F(\overline{\mathbf{w}}_{k+1}^{(i)}), & \text{otherwise} \end{cases} \tag{9b$''$}$$

which is in fact adopted in [1], [17], [20]. Hence, (9b) can be seen as an extension of their methods. ∎

An interesting feature of DDGT lies in the way to handle the coupling constraint $\sum_{i=1}^{n} \mathbf{w}_k^{(i)} = \mathbf{d}$. Notice that DDGT is simply initialized such that $\mathbf{w}_0^{(i)} = 0, \forall i \in \mathcal{V}$ and $\sum_{i=1}^{n} \mathbf{s}_0^{(i)} = \mathbf{d}$. By summing (9c) over $i = 1, \ldots, n$, we obtain that $\sum_{i=1}^{n} (\mathbf{w}_k^{(i)} + \mathbf{s}_k^{(i)}) = \sum_{i=1}^{n} (\mathbf{w}_0^{(i)} + \mathbf{s}_0^{(i)}) = \mathbf{d}$. Thus, if $\mathbf{s}_k^{(i)}$ converges to 0, the constraint is satisfied asymptotically, which is essential to the convergence proof of the DDGT.

By strong duality, the convergence of DDGT can be established by showing the convergence of PPG. However, existing results, e.g., [25]–[27], [42] for the convergence of PPG are established only if f_i is strongly convex and Lipschitz smooth. Note that f_i in (6) is often *not* strongly convex due to the introduction of convex conjugate function F_i^*, though F_i in (1) is strongly convex [30]. This is indeed the case if F_i includes exponential term [43] or logarithmic term [44]. Without Lipschitz smoothness for F_i, we can only obtain that f_i is differentiable and $\frac{1}{\mu}$-Lipschitz smooth [45, Theorem 4.2.1], i.e.,

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq \frac{1}{\mu} \|\mathbf{x} - \mathbf{y}\|, \quad \forall i \in \mathcal{V}, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n. \tag{10}$$

Thus, we still need to prove the convergence of PPG for non-strongly convex objective functions f_i. Particularly, a crucial step in the convergence proof of PPG in [25], [26] uses a complicated 3-dimensional matrix whose spectral radius is strictly less than one for a sufficiently small stepsize. Then, PPG converges at a linear rate. This does not work here since the spectral radius of such a matrix cannot be strictly less than one if f_i is not strongly convex. In fact, we cannot expect a linear convergence rate for the non-strongly convex case [28].

Next, we shall prove that PPG converges to a stationary point at a rate of $O(1/k)$ even for *non-convex* objective functions, based on which we show the convergence and evaluate the convergence rate of DDGT.

IV. CONVERGENCE ANALYSIS

In this section, we first establish the convergence result of PPG in (8) for non-convex f_i, which is of independent interest as the existing results on PPG only apply to the strongly convex case. Then, we show the convergence of the DDGT and evaluate the convergence rate for a special case.

A. Convergence Analysis of PPG Without Convexity

Consider PPG given in (8). With a slight abuse of notation, let f_i be a general differentiable function in the rest of this subsection. Denote

$$X_k = \left[\mathbf{x}_k^{(1)}, \dots, \mathbf{x}_k^{(n)} \right]^{\mathsf{T}} \in \mathbb{R}^{n \times m}$$

$$Y_k = \left[\mathbf{y}_k^{(1)}, \dots, \mathbf{y}_k^{(n)} \right]^{\mathsf{T}} \in \mathbb{R}^{n \times m}$$

$$\nabla \mathbf{f}_k = \left[\nabla f_1(\mathbf{x}_k^{(1)}), \dots, \nabla f_n \left(\mathbf{x}_k^{(n)} \right) \right]^{\mathsf{T}} \in \mathbb{R}^{n \times m} \quad (11)$$

and

$$[A]_{ij} = \begin{cases} a_{ij}, & \text{if } (j,i) \in \mathcal{E} \\ 0, & \text{otherwise,} \end{cases} \quad [B]_{ij} = \begin{cases} b_{ij}, & \text{if } (j,i) \in \mathcal{E} \\ 0, & \text{otherwise.} \end{cases}$$

Note that A is row-stochastic and B is column-stochastic. The starting points of all nodes are set to the same point $\mathbf{x}_0$ for simplicity.

Then, (8) can be written in the following compact form

$$X_{k+1} = A(X_k - \alpha Y_k) \quad (12a)$$

$$Y_{k+1} = BY_k + \nabla \mathbf{f}_{k+1} - \nabla \mathbf{f}_k \quad (12b)$$

The convergence result of PPG for non-strongly convex or even non-convex functions are given in the following theorem.

Theorem 1 (Convergence of PPG Without Convexity): Suppose Assumption 2 holds and $f_i, i \in \mathcal{V}$ in (6) is differentiable and L-Lipschitz smooth (c.f. (10)). If the stepsize α is sufficiently small, i.e., α satisfies (28) and (43), then $\{\mathbf{x}_k^{(i)}\}, i \in \mathcal{V}$ generated by (8) satisfies that

$$\frac{1}{k} \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \leq \frac{f(\mathbf{x}_0) - f^\star}{\gamma k} + \frac{3L\alpha^2(L^2 c_0^2 + c_2^2)}{\gamma(1-\theta)k}$$

$$+ \frac{\alpha \left(\sqrt{n} L c_0 + c_2 \right) \left(1 + \sum_{t=1}^{k_0} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \right)}{\gamma(1-\theta)k} \quad (13)$$

where $\bar{\mathbf{x}}_k = \sum_{i=1}^{n} \pi_A^{(i)} \mathbf{x}_k^{(i)}$, π_A is the normalized left Perron vector of A, and $\theta, c_0, c_2, \gamma, k_0$ are positive constants given in (27), (31), (44), (45) of Appendix, respectively.

Moreover, it holds that

$$\frac{1}{k} \sum_{t=1}^{k} \|X_t - \mathbf{1}\bar{\mathbf{x}}_t^{\mathsf{T}}\|_F^2 \leq \frac{2c_0^2}{(1-\theta)k} + \frac{c_1^2 \alpha^2}{k} \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \quad (14)$$

and if f is convex, $f(\bar{\mathbf{x}}_k)$ converges to $f^\star$.

The proof of Theorem 1 is deferred to the Appendix. Theorem 1 shows that PPG converges to a stationary point of f at a rate of $O(1/k)$ for non-convex functions. The order of convergence rate is consistent with the centralized gradient descent algorithm [41]. Generally, the network size n affects the convergence rate in a complicated way since it closely relates to the network topology and the two weighting matrices A and B. If $\sigma_{\mathsf{A}}, \sigma_{\mathsf{B}}, \delta_{\mathsf{AF}}$ and δ_{BF} in Lemmas 2 and 3 of Appendix do not vary with n, which holds, e.g., by setting $A = B$ in some undirected graphs such as complete graphs and star graphs, then it follows from (27), (28), (31) and (44) that $\theta = O(1)$, $\alpha = O(1/\sqrt{n})$, $c_0 \approx O(\sqrt{n})$ and $\gamma \approx O(\alpha)$. Then, (13) ensures a convergence rate $O(n/k)$, which is reasonable since the Lipschitz constant L is defined in terms of *local* objective functions, and the *global* Lipschitz constant generally increases linearly with n, implying a convergence rate $O(n/k)$ even for the centralized gradient descent method [41, Section 6.1].

B. Convergence of the DDGT

We now establish the convergence and quantify the convergence rate of the DDGT.

Theorem 2 (Convergence of the DDGT): Suppose Assumptions 1 and 2 hold. If the stepsize $\alpha > 0$ is smaller than an upper bound given in (28) and (43) with L replaced by $1/\mu$, then $\{\mathbf{w}_k^{(i)}\}, i \in \mathcal{V}$ in Algorithm 1 converges to an optimal point of (1), i.e., $\lim_{k \to \infty} \mathbf{w}_k^{(i)} = \mathbf{w}_i^\star, \forall i \in \mathcal{V}$.

Proof: Under Assumption 1, the strong duality holds between the original problem (1) and its dual problem (6). Recall the relation between the DDGT (9) and PPG (8). We obtain that $f(\bar{\mathbf{x}}_k)$ converges to $f^\star$ by the convexity of the dual problem and Theorem 1, and $f^\star = -F^\star = -L(W^\star, \mathbf{x})$ for any $x \in \mathbb{R}^m$. Moreover,

$$f(\bar{\mathbf{x}}_k) - f^\star = L(W^\star, \bar{\mathbf{x}}_k) - \inf_{W \in \mathcal{W}} L(W, \bar{\mathbf{x}}_k)$$

$$= L(W^\star, \bar{\mathbf{x}}_k) - L(W_k, \bar{\mathbf{x}}_k)$$

$$\geq \partial_W L(W_k, \bar{\mathbf{x}}_k)^{\mathsf{T}}(W^\star - W_k) + \frac{\mu}{2} \|W_k - W^\star\|_F^2$$

$$\geq \frac{\mu}{2} \|W_k - W^\star\|_F^2 \quad (15)$$

where $L(W, \mathbf{x})$ is the Lagrange function in (2), $W^\star = [\mathbf{w}_1^\star, \dots, \mathbf{w}_n^\star]$ and $W_k = [\mathbf{w}_k^{(i)}, \dots, \mathbf{w}_k^{(i)}]$. The first inequality follows from the strong convexity of F by Assumption 1 and the second inequality uses the first-order necessary condition for a constrained minimization problem. The convergence of $\mathbf{w}_k^{(i)}$ is obtained immediately from (15).

The stepsize condition follows from Theorem 1 and the Lipschitz smoothness of the dual function (c.f. (10)). ∎

Remark 4: We note that it is possible to extend the DDGT to time-varying networks [17], since the convergence of the DDGT

essentially depends on that of PPG, and a recent work [27] shows the feasibility of PPG over time-varying networks for strongly convex functions.

C. Convergence Rate of the DDGT

As in [4] and [10], this subsection focuses on the special case that $\mathcal{W}_i = \mathbb{R}^m$ and F_i is differentiable for all $i \in \mathcal{V}$ for the convergence rate characterization, since the constrained case involves more complicated concepts and notations such as subdifferential.

Under Assumption 1, it follows from [30] that the Karush-Kuhn-Tucker (KKT) condition of (1)

$$\nabla F_1(\mathbf{w}_1^\star) = \cdots = \nabla F_n(\mathbf{w}_n^\star), \tag{16a}$$

$$\sum_{i=1}^{n} \mathbf{w}_i^\star = \mathbf{d} \tag{16b}$$

is a necessary and sufficient condition for optimality. The convergence rate of the DDGT is in terms of (16).

Theorem 3 (Convergence Rate of the DDGT): Suppose that $\mathcal{W}_i = \mathbb{R}^m$, F_i is differentiable for all i, and the conditions in Theorem 2 are satisfied. Let $\overline{\nabla F_k} = \frac{1}{n} \sum_{i=1}^n \nabla F_i(\mathbf{w}_k^{(i)})$, then $\{\mathbf{w}_k^{(i)}\}$ generated by the DDGT satisfies that

$$\frac{1}{k} \sum_{t=1}^{k} \left(\sum_{i=1}^{n} \|\nabla F_i(\mathbf{w}_t^{(i)}) - \overline{\nabla F_t}\|^2 + \|\sum_{i=1}^{n} \mathbf{w}_t^{(i)} - \mathbf{d}\|^2 \right)$$

$$\leq \frac{2(f(\mathbf{x}_0) - f^\star)}{\gamma k} + \frac{6L\alpha^2(L^2 c_0^2 + c_2^2)}{\gamma(1-\theta)k} + \frac{4nc_0^2(\mu^2+1)}{\mu^2(1-\theta)k}$$

$$+ \frac{2\alpha \left(\sqrt{n}Lc_0 + c_2\right)\left(1 + \sum_{t=1}^{k_0} \|\nabla f(\bar{\mathbf{x}}_t)\|^2\right)}{\gamma(1-\theta)k} + O\left(\frac{1}{k^2}\right)$$

where all the constants are defined in Theorem 1.

Moreover, if $F_i, i \in \mathcal{V}$ has Lipschitz continuous gradients, then $\sum_{i=1}^n \|\mathbf{w}_k^{(i)} - \mathbf{w}_i^\star\|^2$ converges linearly, i.e., $\sum_{i=1}^n \|\mathbf{w}_k^{(i)} - \mathbf{w}_i^\star\|^2 \leq O(\lambda^k)$ for some $\lambda \in (0,1)$.

Proof: Since $\mathcal{W}_i = \mathbb{R}^m$, it follows from (9b) that $\mathbf{x}_{k+1}^{(i)} = -\nabla F_i(\mathbf{w}_{k+1}^{(i)})$. Thus,

$$\sum_{i=1}^{n} \left\| \nabla F_i\left(\mathbf{w}_k^{(i)}\right) - \overline{\nabla F_k} \right\|^2$$

$$= \sum_{i=1}^{n} \left\| \mathbf{x}_k^{(i)} - \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_k^{(i)} \right\|^2 = \left\| \left(I - \frac{1}{n}\mathbf{1}\mathbf{1}^\mathsf{T}\right) X_k \right\|_F^2$$

$$\leq 2\| (I - \mathbf{1}\pi_A^\mathsf{T}) X_k \|_F^2 + 2\left\| \left(\mathbf{1}\pi_A^\mathsf{T} - \frac{1}{n}\mathbf{1}\mathbf{1}^\mathsf{T}\right) X_k \right\|_F^2$$

$$= 2\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F^2 + 2\left\| \left(\frac{1}{n}\mathbf{1}\mathbf{1}^\mathsf{T} - \mathbf{1}\pi_A^\mathsf{T}\right)\left(X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\right) \right\|_F^2$$

$$\leq 2n\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F^2 \tag{17}$$

where X_k is defined in (11), $\bar{\mathbf{x}}_k$ and π_A are defined in Theorem 1. The first inequality uses the relation $\|\mathbf{a} + \mathbf{b}\|^2 \leq 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$, and the last inequality follows from $\|\frac{1}{n}\mathbf{1}\mathbf{1}^\mathsf{T} - \mathbf{1}\pi_A^\mathsf{T}\|_F^2 \leq n - 1$.

On the other hand, it follows from (7) and (9b) that

$$\sum_{i=1}^{n} \mathbf{w}_k^{(i)} - \mathbf{d} = -\sum_{i=1}^{n} \nabla f_i\left(\mathbf{x}_k^{(i)}\right)$$

$$= -\left(\nabla f(\bar{\mathbf{x}}_k) + \sum_{i=1}^{n} \left(\nabla f_i(\mathbf{x}_k^{(i)}) - \nabla f_i(\bar{\mathbf{x}}_k)\right) \right)$$

Taking the norm on both sides yields that

$$\left\| \sum_{i=1}^{n} \mathbf{w}_k^{(i)} - \mathbf{d} \right\|^2$$

$$\leq 2\|\nabla f(\bar{\mathbf{x}}_k)\|^2 + 2n \sum_{i=1}^{n} \|\nabla f_i(\mathbf{x}_k^{(i)}) - \nabla f_i(\bar{\mathbf{x}}_k)\|^2$$

$$\leq 2\|\nabla f(\bar{\mathbf{x}}_k)\|^2 + \frac{2n}{\mu^2} \sum_{i=1}^{n} \|\mathbf{x}_k^{(i)} - \bar{\mathbf{x}}_k\|^2$$

$$= 2\|\nabla f(\bar{\mathbf{x}}_k)\|^2 + \frac{2n}{\mu^2}\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F^2 \tag{18}$$

where we use $\|\mathbf{a} + \mathbf{b}\|^2 \leq 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$ again and the Cauchy-Schwarz inequality to obtain the first inequality, and the second inequality follows from (10). Combining (17) and (18) implies that

$$\sum_{i=1}^{n} \left\| \nabla F\left(\mathbf{w}_k^{(i)}\right) - \overline{\nabla F_k} \right\|^2 + \left\| \sum_{i=1}^{n} \mathbf{w}_k^{(i)} - \mathbf{d} \right\|^2$$

$$\leq 2\|\nabla f(\bar{\mathbf{x}}_k)\|^2 + \frac{2n(1+\mu^2)}{\mu^2}\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F^2$$

The desired result then follows from Theorem 1.

The linear convergence rate in the presence of Lipschitz smoothness can be similarly obtained by following the linear convergence of PPG for strongly convex and Lipschitz smooth objective functions ([26, Theorem 1] or [25, Theorem 1]), which is omitted to save space. ∎

Theorem 3 shows that the DDGT converges at a sublinear rate $O(1/k)$ for strongly convex objective functions, and achieves a linear convergence rate if Lipschitz smoothness is further satisfied. In view of Theorem 1, the explicit form of the term corresponding to $O(1/k^2)$ in Theorem 3 can be obtained after tedious computations.

V. NUMERICAL EXPERIMENTS

This section validates our theoretical results and compares the DDGT with existing algorithms via simulation. More precisely, we compare the DDGT with the algorithms in [17], [24] and [10, Mirror-Push-DIGing]. Note that [10] does not provide convergence guarantee for Mirror-Push-DIGing, [17] has no convergence rate results, and [24] only shows the convergence rate for strongly convex and Lipschitz smooth cost functions. Moreover, the algorithm in [24] involves solving a subproblem similar to (9b) per iteration and [17] adopts the update in (9b″) which is a special case of (9b), and hence the computational complexities of the two algorithms are similar to DDGT per iteration. In contrast, Mirror-Push-DIGing [10]
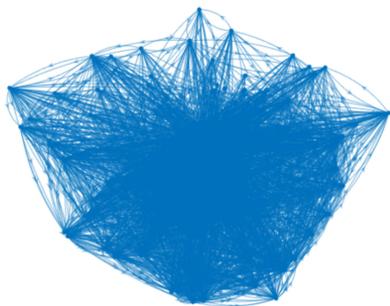
Fig. 1. The communication network in [46] and [47].
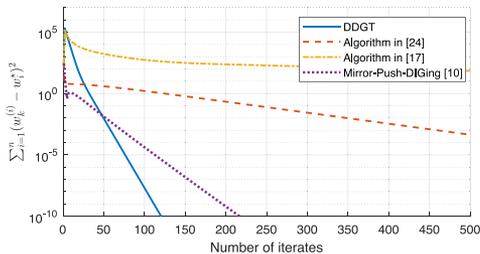
Fig. 2. Convergence rate w.r.t the number of iterations of different algorithms with quadratic cost function $F_i(w_i) = a_i(w_i - b_i)^2$.

Fig. 3. Convergence rate w.r.t the number of iterations of different algorithms with quartic cost function $F_i(w_i) = a_i(w_i - b_i)^2 + c_i(w_i - d_i)^4$.

Fig. 4. Convergence rate w.r.t the number of iterations of different algorithms with quartic cost function $F_i(w_i) = a_i(w_i - b_i)^2 + c_i(w_i - d_i)^4$ and local constraint $-2 \le w_i \le 2, \forall i$.

requires computing a proximal operator, which may have higher computational costs.

We test these algorithms over 126 nodes connected via a directed network, which is a real Email network [46], [47]. Each node i is associated with a local quadratic cost function $F_i(w_i) = a_i(w_i - b_i)^2$ where $a_i \sim \mathcal{U}(0,1)$ and $b_i \sim \mathcal{N}(0,4)$ are randomly sampled. Note that the quadratic cost function is commonly used in the literature [10], [17], [24]. The global constraint is $\sum_{i=1}^{126} w_i = 50$.

We first test the case without local constraints by setting $\mathcal{W}_i = \mathbb{R}^m$. The stepsize used for each algorithm is tuned via a grid search,[1] and all initial conditions are randomly set. Fig. 2 depicts the decay of distance between $\mathbf{w}_k^{(i)}$ and the optimal solution with respect to the number of iterations. It clearly shows that the DDGT has a linear convergence rate and converges faster than algorithms in [17], [24] and [10].

To validate the theoretical result for strongly convex cost functions without Lipschitz smoothness, we test the algorithms with a quartic local cost function $F_i(w_i) = a_i(w_i - b_i)^2 + c_i(w_i - d_i)^4$, where $c_i \sim \mathcal{U}(0,10)$ and $d_i \sim \mathcal{N}(0,4)$ are randomly sampled. Clearly, this function is strongly convex but not Lipschitz smooth. All other settings remain the same and the result is plotted in Fig. 3, where the Mirror-Push-DIGing [10] is not included because its proximal operator is very time-consuming, and an approximate solution for the proximal operator often leads to a poor performance of the algorithm. The dotted line in Fig. 3 is the sequence $\{100/k\}$ with k the number of iterations. We can observe that the convergence rates of all algorithms are slower

---

[1]The grid search scheme works as follows. For each algorithm, we select a "good" stepsize by inspection, and then gradually increase and decrease stepsizes around the selected one with an equal grid size, respectively. Then, we find the fastest one among all the tried stepsizes.
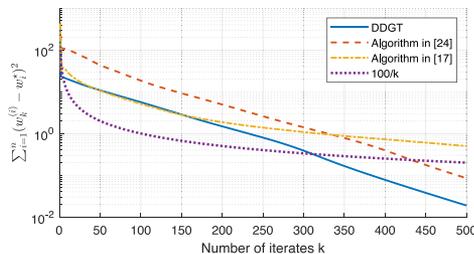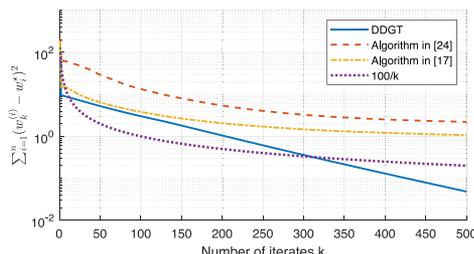
than that in Fig. 2, but the DDGT still outperforms the other two algorithms. Moreover, it is interesting to observe that the DDGT and the algorithm in [24] have near-linear convergence rates, though the theoretical convergence rate for the DDGT is $O(1/k)$.

Finally, we study the effect of local constraints on the convergence rate. To this end, we assign each node a local constraint $-2 \le w_i \le 2$, and test all algorithms with the setting of Fig. 3. The result is depicted in Fig. 4, which shows that the convergence of the DDGT is essentially not affected, while the algorithm in [24] is heavily slowed compared with that in Fig. 3.

VI. CONCLUSION

We proposed the DDGT for distributed resource allocation problems (DRAPs) over directed unbalanced networks. Convergence results are provided by exploiting the strong duality of DRAPs and distributed optimization problems, and taking advantage of the PPG algorithm. We studied the convergence and convergence rate of PPG for non-convex problems and obtained that the DDGT converges linearly for strongly convex and Lipschitz smooth objective functions, and sub-linearly without the Lipschitz smoothness. Future works are to provide tighter bounds for the convergence rate, design asynchronous versions [37], [38], study quantized communication [48], and design accelerated algorithms [49]. In particular, an interesting idea to accelerate the DDGT is to add a vanishing strongly convex regularization term to the dual problems of DRAPs, which may allow a larger stepsize in the early stage and hence possibly lead to faster convergence.

APPENDIX

A. Preliminary Results on Stochastic Matrices

We first introduce three lemmas from [25], [26].

Lemma 1 ([26], [42]): Suppose Assumption 2 holds. The matrix A has a unique unit nonnegative left eigenvector π_A w.r.t. eigenvalue 1, i.e., $\pi_A^\mathsf{T} A = \pi_A^\mathsf{T}$ and $\pi_A^\mathsf{T}\mathbf{1} = 1$. The matrix B has a unique unit right eigenvector π_B w.r.t. eigenvalue 1, i.e., $B\pi_B = \pi_B$ and $\pi_B^\mathsf{T}\mathbf{1} = 1$.

The proof of Lemma 1 follows from the Perron-Frobenius theorem and can be found in [25], [26].

Lemma 2 ([50], [25], [26]): Suppose Assumption 2 holds. There exist matrix norms $\|\cdot\|_\mathsf{A}$ and $\|\cdot\|_\mathsf{B}$ such that $\sigma_\mathsf{A} \triangleq \|A - \mathbf{1}\pi_A^\mathsf{T}\|_\mathsf{A} < 1$ and $\sigma_\mathsf{B} \triangleq \|B - \pi_B\mathbf{1}^\mathsf{T}\|_\mathsf{B} < 1$. Moreover, σ_A and σ_B can be arbitrarily close to the second largest absolute value of the eigenvalues of A and B, respectively.

A method to construct such matrix norms can be found in the proof of Lemma 5.6.10 in [50].

Lemma 3 ([25], [26]): There exist constants $\delta_\mathsf{FA}, \delta_\mathsf{AF}, \delta_\mathsf{FB}$ and δ_BF such that for any $X \in \mathbb{R}^{n \times n}$, we have

$$\|X\|_F \le \delta_\mathsf{FA}\|X\|_\mathsf{A}, \ \|X\|_F \le \delta_\mathsf{FB}\|X\|_\mathsf{B}$$
$$\|X\|_\mathsf{A} \le \delta_\mathsf{AF}\|X\|_F, \ \|X\|_\mathsf{B} \le \delta_\mathsf{BF}\|X\|_F$$

Lemma 3 is a direct result of the norm equivalence theorem. If A and B are symmetric, which means the network is undirected, then $\delta_\mathsf{AF} = \delta_\mathsf{BF} = 1$ and $\delta_\mathsf{FA} = \delta_\mathsf{FB} = \sqrt{n}$.

Note that the norm $\|\cdot\|_\mathsf{A}$ defined in Lemma 2 is only for matrices in $\mathbb{R}^{n \times n}$. To facilitate presentation, we slightly abuse the notation and define a *vector* norm $\|\mathbf{x}\|_\mathsf{A} \triangleq \|\frac{1}{\sqrt{n}}\mathbf{x}\mathbf{1}^\mathsf{T}\|_\mathsf{A}$ for any $\mathbf{x} \in \mathbb{R}^n$, where the norm in the right-hand-side is the matrix norm defined in Lemma 2. Then, we have

$$\|M\mathbf{x}\|_\mathsf{A} = \|\frac{1}{\sqrt{n}}M\mathbf{x}\mathbf{1}^\mathsf{T}\|_\mathsf{A} \le \|M\|_\mathsf{A}\left\|\frac{\mathbf{x}\mathbf{1}^\mathsf{T}}{\sqrt{n}}\right\|_\mathsf{A} = \|M\|_\mathsf{A}\|\mathbf{x}\|_\mathsf{A}$$

where the first equality is by definition and the inequality follows from the sub-multiplicativity of matrix norms. Moreover, for any matrix $X = [\mathbf{x}_1, \ldots, \mathbf{x}_m] \in \mathbb{R}^{n \times m}$, define the matrix norm $\|X\|_\mathsf{A} = \sqrt{\sum_{i=1}^m \|\mathbf{x}_i\|_\mathsf{A}^2}$. Recall that m is the dimension of $\mathbf{x}$ and hence the definition is distinguished from that in Lemma 2. We have

$$\|MX\|_\mathsf{A} = \|[M\mathbf{x}_1, \ldots, M\mathbf{x}_n]\|_\mathsf{A} = \sqrt{\sum_{i=1}^m \|M\mathbf{x}_i\|_\mathsf{A}^2}$$
$$\le \sqrt{\sum_{i=1}^m \|M\|_\mathsf{A}^2\|\mathbf{x}_i\|_\mathsf{A}^2} = \|M\|_\mathsf{A}\|X\|_\mathsf{A}.$$

Therefore, for any $M \in \mathbb{R}^{n \times n}$, $X \in \mathbb{R}^{n \times m}$, and $\mathbf{x} \in \mathbb{R}^n$, the following relation holds

$$\|MX\|_\mathsf{A} \le \|M\|_\mathsf{A}\|X\|_\mathsf{A}, \ \|M\mathbf{x}\|_\mathsf{A} \le \|M\|_\mathsf{A}\|\mathbf{x}\|_\mathsf{A}. \quad (19)$$

Similarly, we can obtain such a relation based on the matrix norm $\|\cdot\|_\mathsf{B}$ defined in Lemma 2.

Next, we define three important auxiliary variables:

$$\bar{\mathbf{x}}_k \triangleq X_k^\mathsf{T}\pi_A, \ \bar{\mathbf{y}}_k \triangleq Y_k^\mathsf{T}\pi_A, \ \hat{\mathbf{y}}_k \triangleq Y_k^\mathsf{T}\mathbf{1} \overset{(12b)}{=} \nabla\mathbf{f}_k^\mathsf{T}\mathbf{1} \quad (20)$$

where $\bar{\mathbf{x}}_k$ is a weighted average of $\mathbf{x}_k^{(i)}$ that is identical to the one defined in Theorem 1, $\bar{\mathbf{y}}_k$ is a weighted average of $\mathbf{y}_k^{(i)}$, and $\hat{\mathbf{y}}_k$ is the sum of $\mathbf{y}_k^{(i)}$.

Finally, for any $X = [\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}]^\mathsf{T} \in \mathbb{R}^{n \times m}$, let

$$\nabla\mathbf{f}(X) = [\nabla f_1(\mathbf{x}^{(1)}), \ldots, \nabla f_n(\mathbf{x}^{(n)})]^\mathsf{T} \in \mathbb{R}^{n \times m},$$

and let $\rho(X)$ denote the spectral radius of matrix X.

B. Proof of Theorem 1

Step 1: Bound $\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_\mathsf{A}$ and $\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{B}$

It follows from (9) that

$$\|X_{k+1} - \mathbf{1}\bar{\mathbf{x}}_{k+1}^\mathsf{T}\|_\mathsf{A}$$
$$= \|AX_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T} - \alpha\left(A - \mathbf{1}\pi_A^\mathsf{T}\right)Y_k\|_\mathsf{A}$$
$$= \left\|\left(A - \mathbf{1}\pi_A^\mathsf{T}\right)\left[\left(X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\right) - \alpha\left(Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\right) - \alpha\pi_B\hat{\mathbf{y}}_k^\mathsf{T}\right]\right\|_\mathsf{A}$$
$$\le \sigma_\mathsf{A}\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_\mathsf{A} + \alpha\sigma_\mathsf{A}\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{A} + \alpha\sigma_\mathsf{A}\|\pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{A}$$
$$\le \sigma_\mathsf{A}\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_\mathsf{A} + \alpha\sigma_\mathsf{A}\delta_\mathsf{AF}\delta_\mathsf{FB}\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{B}$$
$$+ \alpha\sigma_\mathsf{A}\delta_\mathsf{AF}\|\frac{1}{n}\mathbf{1}^\mathsf{T}(\nabla\mathbf{f}(X_k) - \nabla\mathbf{f}\left(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\right) + \nabla\mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}))\|$$
$$\le \alpha\sigma_\mathsf{A}\delta_\mathsf{AF}\delta_\mathsf{FB}\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{B} + \frac{\alpha\sigma_\mathsf{A}\delta_\mathsf{AF}}{n}\|\nabla f(\bar{\mathbf{x}}_k)\|$$
$$+ \left(\sigma_\mathsf{A} + \frac{\alpha\sigma_\mathsf{A}\delta_\mathsf{AF}L}{\sqrt{n}}\right)\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_\mathsf{A} \quad (21)$$

where we use Lemma 2 and (19) to obtain the first inequality, the second inequality is from Lemma 3 and (20), and the last inequality follows from the L-Lipschitz smoothness.

Now we bound $\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{B}$. From (12) we have

$$\|Y_{k+1} - \pi_B\hat{\mathbf{y}}_{k+1}^\mathsf{T}\|_\mathsf{B}$$
$$= \|BY_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T} + (\nabla\mathbf{f}_{k+1} - \nabla\mathbf{f}_k) - (\pi_B\hat{\mathbf{y}}_{k+1}^\mathsf{T} - \pi_B\hat{\mathbf{y}}_k^\mathsf{T})\|_\mathsf{B}$$
$$= \|(B - \pi_B\mathbf{1}^\mathsf{T})(Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}) + (I - \pi_B\mathbf{1}^\mathsf{T})(\nabla\mathbf{f}_{k+1} - \nabla\mathbf{f}_k)\|_\mathsf{B}$$
$$\le \sigma_\mathsf{B}\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{B} + L\delta_\mathsf{BF}\|I - \pi_B\mathbf{1}^\mathsf{T}\|_\mathsf{B}\|X_{k+1} - X_k\|_F$$
$$\le \sigma_\mathsf{B}\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{B} + L\delta_\mathsf{BF}\|X_{k+1} - X_k\|_F. \quad (22)$$

where the last inequality follows from $\|I - \pi_B\mathbf{1}^\mathsf{T}\|_\mathsf{B} = 1$, which can be readily obtained from the construction of the norm $\|\cdot\|_\mathsf{B}$ [50, Lemma 5.6.10]. Moreover, it follows from (12a) that

$$\|X_{k+1} - X_k\|_F = \|AX_k - X_k - \alpha AY_k\|_F$$
$$= \|(A - I)(X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}) - \alpha AY_k\|_F$$
$$\le \|A - I\|\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F + \alpha\|A(Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T} + \pi_B\hat{\mathbf{y}}_k^\mathsf{T})\|_F$$
$$\le 2\sqrt{n}\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F + \alpha\|A\|(\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_F + \|\pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_F)$$
$$\le 2\sqrt{n}\delta_\mathsf{FA}\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_\mathsf{A} + \alpha\sqrt{n}(\delta_\mathsf{FB}\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_\mathsf{B} + \|\hat{\mathbf{y}}_k^\mathsf{T}\|)$$
$$\le 2\sqrt{n}\delta_\mathsf{FA}\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_\mathsf{A} + \alpha\sqrt{n}\delta_\mathsf{FB}\|Y_k - \pi_B\hat{\mathbf{y}}_k\|_\mathsf{B}$$
$$+ \alpha\sqrt{n}\delta_\mathsf{FB}\|\frac{1}{n}\mathbf{1}^\mathsf{T}(\nabla\mathbf{f}(X_k) - \nabla\mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}) + \nabla\mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}))\|$$
$$\le (\alpha\delta_\mathsf{FB}L + 2\sqrt{n}\delta_\mathsf{FA})\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_\mathsf{A}$$
$$+ \alpha\sqrt{n}\delta_\mathsf{FB}\|Y_k - \pi_B\hat{\mathbf{y}}_k\|_\mathsf{B} + \frac{\alpha\delta_\mathsf{FB}}{\sqrt{n}}\|\nabla f(\bar{\mathbf{x}}_k)\|$$

where we used $\|A\| \leq \sqrt{n}$. The above relation combined with (22) yields

$$
\|Y_{k+1} - \pi_B \hat{\mathbf{y}}_{k+1}^{\mathsf{T}}\|_{\mathsf{B}}
$$
$$
\leq (\sigma_{\mathsf{B}} + L\alpha\sqrt{n}\delta_{\mathsf{BF}}\delta_{\mathsf{FB}})\|Y_k - \pi_B \hat{\mathbf{y}}_k^{\mathsf{T}}\|_{\mathsf{B}}
$$
$$
+ (2L\sqrt{n}\delta_{\mathsf{BF}}\delta_{\mathsf{FA}} + L^2\alpha\delta_{\mathsf{BF}}\delta_{\mathsf{FB}})\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^{\mathsf{T}}\|_{\mathsf{A}}
$$
$$
+ \frac{L\alpha\delta_{\mathsf{FB}}\delta_{\mathsf{BF}}}{\sqrt{n}}\|\nabla f(\bar{\mathbf{x}}_k)\|. \tag{23}
$$

Combing (21) and (23) implies the following linear matrix inequality

$$
\underbrace{\begin{bmatrix} \|X_{k+1} - \mathbf{1}\bar{\mathbf{x}}_{k+1}^{\mathsf{T}}\|_{\mathsf{A}} \\ \|Y_{k+1} - \pi_B \hat{\mathbf{y}}_{k+1}^{\mathsf{T}}\|_{\mathsf{B}} \end{bmatrix}}_{\triangleq \mathbf{z}_{k+1}} \preccurlyeq \underbrace{\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}}_{\triangleq P} \underbrace{\begin{bmatrix} \|X_k - \mathbf{1}\bar{\mathbf{x}}_k^{\mathsf{T}}\|_{\mathsf{A}} \\ \|Y_k - \pi_B \hat{\mathbf{y}}_k^{\mathsf{T}}\|_{\mathsf{B}} \end{bmatrix}}_{\triangleq \mathbf{z}_k}
$$
$$
+ \frac{\alpha}{n} \underbrace{\begin{bmatrix} \sigma_{\mathsf{A}}\delta_{\mathsf{AF}}\|\nabla f(\bar{\mathbf{x}}_k)\| \\ L\delta_{\mathsf{BF}}\delta_{\mathsf{FB}}\sqrt{n}\|\nabla f(\bar{\mathbf{x}}_k)\| \end{bmatrix}}_{\triangleq \mathbf{u}_k} \tag{24}
$$

where $\preccurlyeq$ denotes the element-wise less than or equal sign and

$$
P_{11} = \sigma_{\mathsf{A}} + \frac{L\alpha\sigma_{\mathsf{A}}\delta_{\mathsf{AF}}}{\sqrt{n}}, \quad P_{12} = \alpha\sigma_{\mathsf{A}}\delta_{\mathsf{AF}}\delta_{\mathsf{FB}}
$$

$$
P_{21} = L\delta_{\mathsf{BF}}(\alpha\delta_{\mathsf{FB}}L + 2\sqrt{n}\delta_{\mathsf{FA}}), \quad P_{22} = \sigma_{\mathsf{B}} + L\alpha\sqrt{n}\delta_{\mathsf{BF}}\delta_{\mathsf{FB}}
$$

Note that $\rho(P) < 1$ for sufficiently small α, since

$$
\lim_{\alpha \to 0} P = \begin{bmatrix} \sigma_{\mathsf{A}} & 0 \\ 2L\sqrt{n}\delta_{\mathsf{BF}}\delta_{\mathsf{FA}} & \sigma_{\mathsf{B}} \end{bmatrix}
$$

has spectral radius smaller than 1.

The linear matrix inequality (24) implies that

$$
\mathbf{z}_k \preccurlyeq P^{k-1}\mathbf{z}_1 + \sum_{t=1}^{k-1} P^{t-1}\mathbf{u}_{k-t}. \tag{25}
$$

Let θ_1 and θ_2 be the two eigenvalues of P such that $|\theta_2| > |\theta_1|$, and $\theta \triangleq \rho(P) = |\theta_2|$, then P can be diagonalized as

$$
P = T\Lambda T^{-1}, \quad \Lambda = \begin{bmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{bmatrix}. \tag{26}
$$

Let $\Psi = \sqrt{(P_{11} - P_{22})^2 + 4P_{12}P_{21}}$. Note that the analysis so far holds if σ_{A} is replaced by any value in $(\sigma_{\mathsf{A}}, 1)$ (similar for σ_{B}), and hence we assume without loss of generality that $\sigma_{\mathsf{A}} \neq \sigma_{\mathsf{B}}$ to simplify presentation. In that case, Ψ is lower bounded by some positive value that is independent of α, say $\underline{\Psi}$. With some tedious calculations, we have

$$
\theta_1 = \frac{P_{11} + P_{22} - \Psi}{2}
$$

$$
\theta = \theta_2 = \frac{P_{11} + P_{22} + \Psi}{2}
$$
$$
= \frac{1}{2}\left(\sigma_{\mathsf{A}} + \sigma_{\mathsf{B}} + \frac{L\alpha\sigma_{\mathsf{A}}\delta_{\mathsf{AF}}}{\sqrt{n}} + L\alpha\sqrt{n}\delta_{\mathsf{BF}}\delta_{\mathsf{FB}} + \Psi \right). \tag{27}
$$

By letting $\theta = \theta_2 < 1$, we obtain that α should satisfy

$$
\alpha < \frac{(1 - \sigma_{\mathsf{A}})(1 - \sigma_{\mathsf{B}})}{L\delta_{\mathsf{FB}}\delta_{\mathsf{BF}}(2\sigma_{\mathsf{A}}\delta_{\mathsf{AF}}\delta_{\mathsf{FB}} + (2\sigma_{\mathsf{A}}\delta_{\mathsf{AF}}\delta_{\mathsf{FA}} + 1)\sqrt{n})}. \tag{28}
$$

Moreover, T and T^{-1} in (26) can be expressed in an explicit form

$$
T = \begin{bmatrix} \frac{P_{11} - P_{22} - \Psi}{2P_{21}} & \frac{P_{11} - P_{22} + \Psi}{2P_{21}} \\ 1 & 1 \end{bmatrix}, T^{-1} = \begin{bmatrix} -\frac{P_{21}}{\Psi} & \frac{P_{11} - P_{22} + \Psi}{2\Psi} \\ \frac{P_{21}}{\Psi} & \frac{P_{22} - P_{11} + \Psi}{2\Psi} \end{bmatrix}
$$

It then follows from (26) that

$$
0 \preccurlyeq P^k = T\Lambda^k T^{-1}
$$
$$
= \begin{bmatrix} \frac{\theta_1^k + \theta_2^k}{2} + \frac{(P_{11} - P_{22})(\theta_2^k - \theta_1^k)}{2\Psi} & \frac{P_{12}}{\Psi}(\theta_2^k - \theta_1^k) \\ \frac{P_{21}}{\Psi}(\theta_2^k - \theta_1^k) & \frac{\theta_1^k + \theta_2^k}{2} + \frac{(P_{11} - P_{22})(\theta_1^k - \theta_2^k)}{2\Psi} \end{bmatrix}
$$
$$
\preccurlyeq \theta^k \begin{bmatrix} 1 & 1/\underline{\Psi} \\ (2L\sqrt{n}\delta_{\mathsf{BF}}\delta_{\mathsf{FA}} + L)/\underline{\Psi} & 1 \end{bmatrix} \tag{29}
$$

where we used $|P_{11} - P_{22}| \leq \Psi$, $\Psi \geq \underline{\Psi}$, and the bound (28) to obtain the inequality.

Combining (24), (25) and (29) yields that

$$
\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^{\mathsf{T}}\|_F \leq c_0\theta^{k-1} + c_1\alpha\sum_{t=1}^{k-1}\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_{k-t})\|
$$

$$
\|Y_k - \pi_B \hat{\mathbf{y}}_k^{\mathsf{T}}\|_F \leq c_2\theta^{k-1} + c_3\alpha\sum_{t=1}^{k-1}\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_{k-t})\| \tag{30}
$$

where c_0, c_1, c_2 and c_3 are constants given as follows

$$
c_0 = \|Y_0 - \pi_B \hat{\mathbf{y}}_0^{\mathsf{T}}\|_{\mathsf{B}}/\underline{\Psi} \leq \frac{\delta_{\mathsf{BF}}}{\underline{\Psi}}\left(\sum_{i=1}^n \|\nabla f_i(\mathbf{x}_0)\|^2 \right)^{1/2}
$$

$$
c_1 = \frac{\sigma_{\mathsf{A}}\delta_{\mathsf{AF}}}{n} + \frac{L\delta_{\mathsf{BF}}\delta_{\mathsf{FB}}}{\sqrt{n}\underline{\Psi}}
$$

$$
c_2 = \|Y_0 - \pi_B \hat{\mathbf{y}}_0^{\mathsf{T}}\|_{\mathsf{B}} \leq \delta_{\mathsf{BF}}\left(\sum_{i=1}^n \|\nabla f_i(\mathbf{x}_0)\|^2 \right)^{1/2}
$$

$$
c_3 = \frac{2L\sqrt{n}\delta_{\mathsf{BF}}\delta_{\mathsf{FA}} + L}{n\underline{\Psi}} + \frac{L\delta_{\mathsf{BF}}\delta_{\mathsf{FB}}}{\sqrt{n}}. \tag{31}
$$

Step 2: Bound $\|\bar{\mathbf{y}}_k\|^2$

From (12) and the L-Lipschitz smoothness, we have

$$
f(\bar{\mathbf{x}}_{k+1}) \leq f(\bar{\mathbf{x}}_k) - \alpha\nabla f(\bar{\mathbf{x}}_k)^{\mathsf{T}}\bar{\mathbf{y}}_k + \frac{L\alpha^2}{2}\|\bar{\mathbf{y}}_k\|^2. \tag{32}
$$

Note that

$$
\bar{\mathbf{y}}_k = Y_k^{\mathsf{T}}\pi_A = (Y_k - \pi_B\hat{\mathbf{y}}_k^{\mathsf{T}} + \pi_B\hat{\mathbf{y}}_k^{\mathsf{T}})^{\mathsf{T}}\pi_A
$$
$$
= (Y_k - \pi_B\hat{\mathbf{y}}_k^{\mathsf{T}})^{\mathsf{T}}\pi_A + Y_k^{\mathsf{T}}\mathbf{1}\pi_B^{\mathsf{T}}\pi_A
$$
$$
= (Y_k - \pi_B\hat{\mathbf{y}}_k^{\mathsf{T}})^{\mathsf{T}}\pi_A + (\nabla \mathbf{f}(X_k) - \nabla \mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^{\mathsf{T}}))^{\mathsf{T}}\mathbf{1}\pi_B^{\mathsf{T}}\pi_A
$$
$$
+ \pi_B^{\mathsf{T}}\pi_A\nabla f(\bar{\mathbf{x}}_k) \tag{33}
$$

where we used the relation $Y_k^\mathsf{T}\mathbf{1} = \nabla\mathbf{f}(X_k)^\mathsf{T}\mathbf{1}$ and $\nabla\mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T})^\mathsf{T}\mathbf{1} = \nabla f(\bar{\mathbf{x}}_k)$. Then, we have

$$
\begin{aligned}
&-\nabla f(\bar{\mathbf{x}}_k)^\mathsf{T}\bar{\mathbf{y}}_k \\
&= -\nabla f(\bar{\mathbf{x}}_k)^\mathsf{T}(\nabla\mathbf{f}(X_k) - \nabla\mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}))^\mathsf{T}\mathbf{1}\pi_B^\mathsf{T}\pi_A \\
&\quad -\nabla f(\bar{\mathbf{x}}_k)^\mathsf{T}(Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T})^\mathsf{T}\pi_A - \pi_B^\mathsf{T}\pi_A\|\nabla f(\bar{\mathbf{x}}_k)\|^2 \\
&\leq -\pi_B^\mathsf{T}\pi_A\|\nabla f(\bar{\mathbf{x}}_k)\|^2 + L\sqrt{n}\|\nabla f(\bar{\mathbf{x}}_k)\|\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F \\
&\quad + \|\nabla f(\bar{\mathbf{x}}_k)\|\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|_F \qquad (34)
\end{aligned}
$$

where we used $\|\pi_A\| \leq 1$, and the Lipschitz smoothness $\|\nabla\mathbf{f}(X_k) - \nabla\mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T})\|_F \leq L\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F$ to obtain the last inequality.

Moreover, it follows from (33) and the relation $\|\mathbf{a} + \mathbf{b} + \mathbf{c}\|^2 \leq 3\|\mathbf{a}\|^2 + 3\|\mathbf{b}\|^2 + 3\|\mathbf{c}\|^2$ that

$$
\begin{aligned}
\|\bar{\mathbf{y}}_k\|^2 &\leq 3\|(Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T})^\mathsf{T}\pi_A\|^2 + 3\|\pi_B^\mathsf{T}\pi_A\nabla f(\bar{\mathbf{x}}_k)\|^2 \\
&\quad + 3\|(\nabla\mathbf{f}(X_k) - \nabla\mathbf{f}(\mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}))^\mathsf{T}\mathbf{1}\pi_B^\mathsf{T}\pi_A\|^2 \\
&\leq 3\|Y_k - \pi_B\hat{\mathbf{y}}_k^\mathsf{T}\|^2 + 3(\pi_B^\mathsf{T}\pi_A)^2\|\nabla f(\bar{\mathbf{x}}_k)\|^2 \\
&\quad + 3\,L^2 n\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|^2. \qquad (35)
\end{aligned}
$$

Step 3: Bound $\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_t)\|\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F$ **and** $\sum_{t=1}^{k}\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F^2$

We first bound the summation of the terms $\|\nabla f(\bar{\mathbf{x}}_t)\|\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F$ and $\|\nabla f(\bar{\mathbf{x}}_t)\|\|Y_t - \pi_B\hat{\mathbf{y}}_t^\mathsf{T}\|_F$ in (34) over $t = 1, \ldots, k$. It follows from (30) that

$$
\begin{aligned}
&\|\nabla f(\bar{\mathbf{x}}_k)\|\|X_k - \mathbf{1}\bar{\mathbf{x}}_k^\mathsf{T}\|_F \\
&\leq c_0\theta^{k-1}\|\nabla f(\bar{\mathbf{x}}_k)\| + c_1\alpha\|\nabla f(\bar{\mathbf{x}}_k)\|\sum_{t=1}^{k-1}\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_{k-t})\| \\
&\qquad\qquad (36)
\end{aligned}
$$

Then, define

$$
\vartheta_t = [\theta^{t-2}, \theta^{t-3}, \ldots, \theta, 1, 0, \ldots, 0]^\mathsf{T} \in \mathbb{R}^k
$$

$$
\tilde{\vartheta}_t = [\underbrace{0, \ldots, 0}_{t-1}, 1, 0, \ldots, 0]^\mathsf{T} \in \mathbb{R}^k
$$

$$
\upsilon_k = [\|\nabla f(\bar{\mathbf{x}}_1)\|, \ldots, \|\nabla f(\bar{\mathbf{x}}_k)\|]^\mathsf{T} \in \mathbb{R}^k
$$

$$
\tilde{\Theta}_k = \sum_{t=1}^{k}\vartheta_t\tilde{\vartheta}_t^\mathsf{T} = \begin{bmatrix} 0 & 1 & \theta & \cdots & \theta^{k-2} \\ & 0 & 1 & \cdots & \theta^{k-1} \\ & & \ddots & \ddots & \vdots \\ & & & 0 & 1 \\ & & & & 0 \end{bmatrix}
$$

where θ is defined in (27). Note that $\nabla f(\bar{\mathbf{x}}_k) = \upsilon_k^\mathsf{T}\tilde{\vartheta}_t$ and $\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_{k-t})\| = \upsilon_k^\mathsf{T}\vartheta_t$, which combined with the relation $\|\nabla f(\bar{\mathbf{x}}_k)\| \leq 1 + \|\nabla f(\bar{\mathbf{x}}_k)\|^2$ and (36) yields

$$
\begin{aligned}
&\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_t)\|\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F \\
&\leq c_0\sum_{t=1}^{k}\theta^{t-1}(1 + \|\nabla f(\bar{\mathbf{x}}_t)\|^2) + c_1\alpha\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_k)\|\vartheta_t^\mathsf{T}\upsilon_k
\end{aligned}
$$

$$
\begin{aligned}
&\leq \frac{c_0}{1-\theta} + c_0\sum_{t=1}^{k}\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_t)\|^2 + c_1\alpha\sum_{t=1}^{k}\upsilon_k^\mathsf{T}\tilde{\vartheta}_t\vartheta_t^\mathsf{T}\upsilon_k \\
&\leq \frac{c_0}{1-\theta} + c_0\sum_{t=1}^{k}\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_t)\|^2 + c_1\alpha\upsilon_k^\mathsf{T}\tilde{\Theta}_k\upsilon_k. \qquad (37)
\end{aligned}
$$

The last term $\upsilon_k^\mathsf{T}\tilde{\Theta}_k\upsilon_k$ in (37) can be bounded by

$$
\upsilon_k^\mathsf{T}\tilde{\Theta}_k\upsilon_k = \upsilon_k^\mathsf{T}\frac{\tilde{\Theta}_k + \tilde{\Theta}_k^\mathsf{T}}{2}\upsilon_k \leq \frac{1}{2}\rho(\tilde{\Theta}_k + \tilde{\Theta}_k^\mathsf{T})\|\upsilon_k\|^2 \leq \frac{\|\upsilon_k\|^2}{1-\theta}
$$

where the last inequality follows from $\rho(\tilde{\Theta}_k + \tilde{\Theta}_k^\mathsf{T}) \leq \|\tilde{\Theta}_k + \tilde{\Theta}_k^\mathsf{T}\|_1 \leq \|\tilde{\Theta}_k\|_1 + \|\tilde{\Theta}_k\|_\infty \leq \frac{2}{1-\theta}$. Thus, we have from (37) that

$$
\begin{aligned}
&\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_t)\|\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F \\
&\leq \frac{c_0}{1-\theta} + c_0\sum_{t=1}^{k}\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_t)\|^2 + \frac{c_1\alpha}{1-\theta}\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_t)\|^2 \\
&\qquad\qquad (38)
\end{aligned}
$$

Similarly, we can bound $\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_t)\|\|Y_t - \pi_B\hat{\mathbf{y}}_t^\mathsf{T}\|_F$ as follows,

$$
\begin{aligned}
&\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_t)\|\|Y_t - \pi_B\hat{\mathbf{y}}_t^\mathsf{T}\|_F \\
&\leq \frac{c_2}{1-\theta} + c_2\sum_{t=1}^{k}\theta^{t-1}\|\nabla f(\bar{\mathbf{x}}_t)\|^2 + \frac{c_3\alpha}{1-\theta}\sum_{t=1}^{k}\|\nabla f(\bar{\mathbf{x}}_t)\|^2.
\end{aligned}
$$

Next, we bound $\sum_{t=1}^{k}\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F^2$ and $\sum_{t=1}^{k}\|Y_t - \pi_B\hat{\mathbf{y}}_t^\mathsf{T}\|_F^2$. We first consider $\sum_{t=1}^{k}\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F^2$. For any $k \in \mathbb{N}$, define

$$
\nu_k = [c_0, c_1\alpha\|\nabla f(\bar{\mathbf{x}}_1)\|, \ldots, c_1\alpha\|\nabla f(\bar{\mathbf{x}}_{k-1})\|]^\mathsf{T} \in \mathbb{R}^k
$$

$$
\phi_t = [\theta^{t-1}, \theta^{t-2}, \ldots, \theta, 1, 0, \ldots, 0]^\mathsf{T} \in \mathbb{R}^k
$$

$$
\Theta_k = \sum_{t=1}^{k}\phi_t\phi_t^\mathsf{T} \in \mathbb{R}^{k\times k}
$$

where the elements are defined in (27) and (31). Clearly, Θ_k is nonnegative and positive semi-definite. We have from (30) that $\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F \leq \nu_k^\mathsf{T}\phi_t$, and hence

$$
\sum_{t=1}^{k}\|X_t - \mathbf{1}\bar{\mathbf{x}}_t^\mathsf{T}\|_F^2 \leq \nu_k^\mathsf{T}\Theta_k\nu_k \leq \|\Theta_k\|\|\nu_k\|^2. \qquad (39)
$$

To bound $\|\Theta_k\|$, let $[\Theta_k]_{ij}$ be the element in the i-th row and j-th column of Θ_k. For any $0 < i \leq j \leq k$, we have

$$
\begin{aligned}
&[\Theta_k]_{ij} \\
&= \sum_{t=j-1}^{k}\theta^{t-i+1}\theta^{t-j+1} = \sum_{t=j-1}^{k}\theta^{2t-i-j+2} \\
&= \frac{\theta^{2j-2}(1 - \theta^{2(k-j+2)})}{1 - \theta^2}\theta^{2-i-j} = \frac{\theta^{j-i}(1 - \theta^{2(k-j+2)})}{1 - \theta^2}.
\end{aligned}
$$

Since Θ_k is symmetric, $\|\Theta_k\|$ equals to its spectral radius. By invoking the Gershgorin circle theorem, we have

$$
\begin{aligned}
\|\Theta_k\| &\le \max_j \sum_{i=1}^{k} [\Theta_k]_{ij} = \max_j \left[\sum_{i=1}^{j} [\Theta_k]_{ij} + \sum_{i=j+1}^{k} [\Theta_k]_{ij} \right] \\
&= \max_j \left[\sum_{i=1}^{j} [\Theta_k]_{ij} + \sum_{i=j+1}^{k} [\Theta_k]_{ji} \right] \\
&= \max_j \left[\sum_{i=1}^{j} \frac{\theta^{j-i}(1 - \theta^{2(k-j+2)})}{1 - \theta^2} \right. \\
&\quad \left. + \sum_{i=j+1}^{k} \frac{\theta^{i-j}(1 - \theta^{2(k-i+2)})}{1 - \theta^2} \right] \\
&= \max_j \left[\frac{(1 - \theta^j)(1 - \theta^{2(k-j+2)})}{(1-\theta)(1-\theta^2)} \right. \\
&\quad \left. + \frac{\theta(1 - \theta^{k-j}) + \theta^{2(k-j+2)}(1 - \theta^{j-k})}{(1-\theta)(1-\theta^2)} \right] \\
&\stackrel{j=k}{=} \frac{(1+\theta^2)(1-\theta^k)}{1-\theta} \le \frac{2}{1-\theta}.
\end{aligned}
$$

It then follows from (39) that

$$
\begin{aligned}
\sum_{t=1}^{k} \|X_t - \mathbf{1}\bar{\mathbf{x}}_t^{\mathsf{T}}\|_F^2 &\le \frac{2}{1-\theta} \left[c_0^2 + c_1^2 \alpha^2 \sum_{t=1}^{k-1} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \right] \\
\sum_{t=1}^{k} \|Y_t - \pi_B \hat{\mathbf{y}}_t\|_F^2 &\le \frac{2}{1-\theta} \left[c_2^2 + c_3^2 \alpha^2 \sum_{t=1}^{k-1} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \right]
\end{aligned}
$$
(40)

Step 4: Bound $\sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2$

Combining (32), (34) and (35) implies that

$$
\begin{aligned}
f(\bar{\mathbf{x}}_{k+1}) \\
\le f(\bar{\mathbf{x}}_k) &- \alpha \nabla f(\bar{\mathbf{x}}_k)^{\mathsf{T}} \bar{\mathbf{y}}_k + \frac{L\alpha^2}{2} \|\bar{\mathbf{y}}_k\|^2 \\
\le f(\bar{\mathbf{x}}_k) &- \alpha \pi_B^{\mathsf{T}} \pi_A \left(1 - \frac{3L\alpha \pi_B^{\mathsf{T}} \pi_A}{2} \right) \|\nabla f(\bar{\mathbf{x}}_k)\|^2 \\
&+ \alpha \|\nabla f(\bar{\mathbf{x}}_k)\| \|Y_k - \pi_B \hat{\mathbf{y}}_k^{\mathsf{T}}\|_F + \frac{3L\alpha^2}{2} \|Y_k - \pi_B \hat{\mathbf{y}}_k^{\mathsf{T}}\|_F^2 \\
&+ \frac{3\,L^3\alpha^2}{2} \|X_k - \mathbf{1}\bar{\mathbf{x}}_k^{\mathsf{T}}\|_F^2 + L\alpha\sqrt{n} \|\nabla f(\bar{\mathbf{x}}_k)\| \|X_k - \mathbf{1}\bar{\mathbf{x}}_k^{\mathsf{T}}\|_F.
\end{aligned}
$$
(41)

Summing both sides of (41) over $1, \ldots, k$, we have

$$
\begin{aligned}
&\alpha \pi_B^{\mathsf{T}} \pi_A \left(1 - \frac{3L\alpha \pi_B^{\mathsf{T}} \pi_A}{2} \right) \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \\
&\le f(\mathbf{x}_0) - f(\bar{\mathbf{x}}_k) + \alpha \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\| \|Y_t - \pi_B \hat{\mathbf{y}}_t^{\mathsf{T}}\|_F
\end{aligned}
$$

$$
\begin{aligned}
&+ \frac{3L\alpha^2}{2} \sum_{t=1}^{k} \left(\|Y_t - \pi_B \hat{\mathbf{y}}_t\|_F^2 + L^2 \|X_t - \mathbf{1}\bar{\mathbf{x}}_t^{\mathsf{T}}\|_F^2 \right) \\
&+ \sum_{t=1}^{k} L\alpha\sqrt{n} \|\nabla f(\bar{\mathbf{x}}_t)\| \|X_t - \mathbf{1}\bar{\mathbf{x}}_t^{\mathsf{T}}\|_F \\
&\le f(\mathbf{x}_0) - f^{\star} + \frac{3L\alpha^2(L^2 c_0^2 + c_2^2)}{1-\theta} + \frac{\alpha(\sqrt{n}Lc_0 + c_2)}{1-\theta} \\
&+ \frac{3L\alpha^4(L^2 c_1^2 + c_3^2)}{1-\theta} \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \\
&+ \frac{\alpha^2(\sqrt{n}Lc_1 + c_3)}{1-\theta} \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 \\
&+ \alpha(\sqrt{n}Lc_0 + c_2) \sum_{t=1}^{k} \theta^{t-1} \|\nabla f(\bar{\mathbf{x}}_t)\|^2
\end{aligned}
$$
(42)

where the last inequality follows from (38) and (40).

We can move the terms related to $\sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2$ in the right-hand-side of (42) to the left-hand-side to bound $\sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2$. To this end, the stepsize α should satisfy

$$
\alpha < \left(\frac{3L\pi_B^{\mathsf{T}} \pi_A}{2} \right.
$$
$$
\left. + \frac{3\,L^3 c_1^2 + 3\,Lc_3^2 + L\sqrt{n}(c_0 + c_1) + c_2 + c_3}{(1-\theta)\pi_B^{\mathsf{T}} \pi_A} \right)^{-1}
$$
(43)

which is followed by

$$
\begin{aligned}
\gamma \triangleq \alpha \pi_B^{\mathsf{T}} \pi_A &\left(1 - \frac{3L\alpha \pi_B^{\mathsf{T}} \pi_A}{2} \right. \\
&\left. - \frac{\alpha(3\,L^3 c_1^2 + 3\,Lc_3^2 + L\sqrt{n}(c_0 + c_1) + c_2 + c_3)}{(1-\theta)\pi_B^{\mathsf{T}} \pi_A} \right) > 0.
\end{aligned}
$$
(44)

If $\theta^k \le \frac{\alpha}{1-\theta}$, i.e.,

$$
k \ge k_0 \triangleq \frac{\ln(\alpha) - \ln(1-\theta)}{\ln(\theta)},
$$
(45)

then it follows from (42) that

$$
\begin{aligned}
\gamma \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 &\le f(\mathbf{x}_0) - f^{\star} + \frac{3L\alpha^2(L^2 c_0^2 + c_2^2)}{(1-\theta)} \\
&+ \frac{\alpha(\sqrt{n}Lc_0 + c_2)}{(1-\theta)} + \alpha(\sqrt{n}Lc_0 + c_2) \sum_{t=1}^{k_0} \|\nabla f(\bar{\mathbf{x}}_t)\|^2
\end{aligned}
$$

Thus, we have

$$
\begin{aligned}
\frac{1}{k} \sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2 &\le \frac{f(\mathbf{x}_0) - f^{\star}}{\gamma k} + \frac{3L\alpha^2(L^2 c_0^2 + c_2^2)}{\gamma(1-\theta)k} \\
&+ \frac{\alpha(\sqrt{n}Lc_0 + c_2)(1 + \sum_{t=1}^{k_0} \|\nabla f(\bar{\mathbf{x}}_t)\|^2)}{\gamma(1-\theta)k}
\end{aligned}
$$

which is (13) in Theorem 1. The inequality (14) follows from (40) immediately.

Now we look back at (41). Jointly with (13), (38), (40) and (41), it follows from the supermartingale convergence theorem [41, Proposition A.4.4] that $f(\bar{\mathbf{x}}_k)$ converges. If f is further convex, it follows from the convergence of $\sum_{t=1}^{k} \|\nabla f(\bar{\mathbf{x}}_t)\|^2$ that $f(\bar{\mathbf{x}}_k)$ converges to $f^\star$.

ACKNOWLEDGMENT

REFERENCES

[1] S. Yang, S. Tan, and J.-X. Xu, "Consensus based approach for economic dispatch problem in a smart grid," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4416–4426, Nov. 2013.

[2] Y. Ho, L. Servi, and R. Suri, "A class of center-free resource allocation algorithms," *IFAC Proc. Volumes*, vol. 13, no. 6, pp. 475–482, 1980.

[3] L. Xiao and S. Boyd, "Optimal scaling of a gradient method for distributed resource allocation," *J. Optim. Theory Appl.*, vol. 129, no. 3, pp. 469–488, 2006.

[4] H. Lakshmanan and D. P. De Farias, "Decentralized resource allocation in dynamic networks of agents," *SIAM J. Optim.*, vol. 19, no. 2, pp. 911–940, 2008.

[5] C. Zhao, J. Chen, J. He, and P. Cheng, "Privacy-preserving consensus-based energy management in smart grids," *IEEE Trans. Signal Process.*, vol. 66, no. 23, pp. 6162–6176, Dec. 2018.

[6] T. T. Doan and A. Olshevsky, "Distributed resource allocation on dynamic networks in quadratic time," *Syst. Control Lett.*, vol. 99, pp. 57–63, 2017.

[7] T.-H. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 482–497, Jan. 2015.

[8] T.-H. Chang, "A proximal dual consensus ADMM method for multi-agent constrained optimization," *IEEE Trans. Signal Process.*, vol. 64, no. 14, pp. 3719–3734, Jul. 2016.

[9] N. S. Aybat and E. Y. Hamedani, "A distributed ADMM-like method for resource sharing over time-varying networks," *SIAM J. Optim.*, vol. 29, no. 4, pp. 3036–3068, 2019.

[10] A. Nedić, A. Olshevsky, and W. Shi, "Improved convergence rates for distributed resource allocation," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 172–177.

[11] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "A dual splitting approach for distributed resource allocation with regularization," *IEEE Trans. Control Netw. Syst.*, vol. 6, no. 1, pp. 403–414, Mar. 2018.

[12] S. Liang, X. Zeng, G. Chen, and Y. Hong, "Distributed sub-optimal resource allocation via a projected form of singular perturbation," 2019, *arXiv:1906.03628*.

[13] Y. Zhu, W. Ren, W. Yu, and G. Wen, "Distributed resource allocation over directed graphs via continuous-time algorithms," *IEEE Trans. Syst., Man, Cybern.: Syst.*, to be published, doi: 10.1109/TSMC.2019.2894862.

[14] P. Xie, K. You, R. Tempo, S. Song, and C. Wu, "Distributed convex optimization with inequality constraints over time-varying unbalanced digraphs," *IEEE Trans. Autom. Control*, vol. 63, no. 12, pp. 4331–4337, Dec. 2018.

[15] K. Cai and H. Ishii, "Average consensus on general strongly connected digraphs," *Automatica*, vol. 48, no. 11, pp. 2750–2761, 2012.

[16] Y. Xu, K. Cai, T. Han, and Z. Lin, "A fully distributed approach to resource allocation problem under directed and switching topologies," in *Proc. 10th Asian Control Conf.*, 2015, pp. 1–6.

[17] Y. Xu, T. Han, K. Cai, Z. Lin, G. Yan, and M. Fu, "A distributed algorithm for resource allocation over dynamic digraphs," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2600–2612, May 2017.

[18] P. Li and J. Hu, "An ADMM based distributed finite-time algorithm for economic dispatch problems," *IEEE Access*, vol. 6, pp. 30 969–30 976, 2018.

[19] A. Falsone, I. Notarnicola, G. Notarstefano, and M. Prandini, "Tracking-ADMM for distributed constraint-coupled optimization," 2019, *arXiv:1907.10860*.

[20] T. Yang *et al.*, "A distributed algorithm for economic dispatch over time-varying directed networks with delays," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5095–5106, 2017.

[21] X. Shi, Y. Wang, S. Song, and G. Yan, "Distributed optimisation for resource allocation with event-triggered communication over general directed topology," *Int. J. Syst. Sci.*, vol. 49, no. 6, pp. 1119–1130, 2018.

[22] H. Zhang, H. Li, Y. Zhu, Z. Wang, and D. Xia, "A distributed stochastic gradient algorithm for economic dispatch over directed network with communication delays," *Int. J. Elect. Power Energy Syst.*, vol. 110, pp. 759–771, 2019.

[23] Y. Yuan, H. Li, J. Hu, and Z. Wang, "Stochastic gradient-push for economic dispatch on time-varying directed networks with delays," *Int. J. Elect. Power Energy Syst.*, vol. 113, pp. 564–572, 2019.

[24] H. Li, Q. L, and T. Huang, "Convergence analysis of a distributed optimization algorithm with a general unbalanced directed communication network," *IEEE Trans. Netw. Sci. Eng.*, vol. 6, no. 3, pp. 237–248, Jul.–Sep. 2019.

[25] S. Pu, W. Shi, J. Xu, and A. Nedic, "Push-pull gradient methods for distributed optimization in networks," *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2020.2972824.

[26] R. Xin and U. A. Khan, "A linear algorithm for optimization over directed graphs with geometric convergence," *IEEE Control Syst. Lett.*, vol. 2, no. 3, pp. 315–320, Jul. 2018.

[27] F. Saadatniaki, R. Xin, and U. A. Khan, "Decentralized optimization over time-varying directed graphs with row and column-stochastic matrices," *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2020.2969721.

[28] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, vol. 87. Berlin, Germany: Springer, 2013.

[29] B. Gharesifard and J. Cortés, "When does a digraph admit a doubly stochastic adjacency matrix?" in *Proc. IEEE Amer. Control Conf.*, 2010, pp. 2440–2445.

[30] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[31] D. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 2016.

[32] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 601–615, Mar. 2015.

[33] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM J. Optim.*, vol. 27, no. 4, pp. 2597–2633, 2017.

[34] J. Zeng and W. Yin, "Extrapush for convex smooth decentralized optimization over directed networks," *J. Comput. Math.*, vol. 35, no. 4, pp. 383–396, 2017.

[35] C. Xi and U. A. Khan, "DEXTRA: A fast algorithm for optimization over directed graphs," *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 4980–4993, Oct. 2017.

[36] V. S. Mai and E. H. Abed, "Distributed optimization over weighted directed graphs using row stochastic matrix," in *Proc. Amer. Control Conf.*, 2016, pp. 7165–7170.

[37] J. Zhang and K. You, "AsySPA: An exact asynchronous algorithm for convex optimization over digraphs," *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2019.2930234.

[38] J. Zhang and K. You, "Asynchronous decentralized optimization in directed networks," 2019, *arXiv:1901.08215*.

[39] X. Zhao and A. H. Sayed, "Asynchronous adaptation and learning over networks-Part I: Modeling and stability analysis," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 811–826, Feb. 2015.

[40] T. Wu, K. Yuan, Q. Ling, W. Yin, and A. H. Sayed, "Decentralized consensus optimization with asynchrony and delays," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 4, no. 2, pp. 293–307, Jun. 2018.

[41] D. P. Bertsekas, *Convex Optimization Algorithms*. Belmont, MA, USA: Athena Scientific, 2015.

[42] S. Pu and A. Nedić, "A distributed stochastic gradient tracking method," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 963–968.

[43] Y. Zhao, X. He, and L. Chen, "A distributed strategy based on ADMM for dynamic economic dispatch problems considering environmental cost function with exponential term," in *Proc. IECON 43rd Annu. Conf. IEEE Ind. Electron. Soc.*, 2017, pp. 7387–7392.

[44] L. Zheng and C. W. Tan, "Optimal algorithms in wireless utility maximization: Proportional fairness decomposition and nonlinear perron-frobenius theory framework," *IEEE Trans. Wireless Commun.*, vol. 13, no. 4, pp. 2086–2095, Apr. 2014.

[45] J.-B. Hiriart-Urruty and C. Lemaréchal, *Conjugacy in Convex Analysis*. Berlin, Germany: Springer, 1993, pp. 35–90

[46] "Manufacturing emails network dataset – KONECT," Sep. 2016. [Online]. Available: http://konect.uni-koblenz.de/networks/radoslaw_email

[47] J. Kunegis, "KONECT – the koblenz network collection," in *Proc. Int. Conf. World Wide Web Companion*, 2013, pp. 1343–1350. [Online]. Available: http://userpages.uni-koblenz.de/kunegis/paper/kunegis-koblenz-network-collection.pdf

[48] J. Zhang, K. You, and T. Başar, "Distributed discrete-time optimization in multiagent networks using only sign of relative state," *IEEE Trans. Autom. Control*, vol. 64, no. 6, pp. 2352–2367, Jun. 2018.

[49] G. Qu and N. Li, "Accelerated distributed Nesterov gradient descent," *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2019.2937496.

[50] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

Jiaqi Zhang received the B.S. degree in electronic and information engineering from the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China, in 2016. He is currently working toward the Ph.D. degree with the Department of Automation, Tsinghua University, Beijing, China. His research interests include networked control systems, distributed or decentralized optimization, and their applications.

Keyou You (Senior Member, IEEE) received the B.S. degree in statistical science from Sun Yat-sen University, Guangzhou, China, in 2007 and the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University (NTU), Singapore, in 2012. After briefly working as a Research Fellow with NTU, he joined Tsinghua University, Beijing, China, where he is currently a Tenured Associate Professor with the Department of Automation. He held visiting positions with Politecnico di Torino, Hong Kong University of Science and Technology, University of Melbourne, etc. His current research interests include networked control systems, distributed optimization and learning, and their applications. Dr. You is the recipient of the Guan Zhaozhi Award from the 29th Chinese Control Conference in 2010 and the Asian Control Association Temasek Young Educator Award in 2019. He received the National Science Fund for Excellent Young Scholars in 2017. He is currently an Associate Editor for the IEEE TRANSACTIONS ON CYBERNETICS, IEEE CONTROL SYSTEMS LETTERS, and *Systems and Control Letters*.
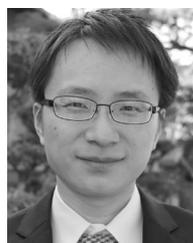
Kai Cai (Senior Member, IEEE) received the B.Eng. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2006, the M.A.Sc. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2008, and the Ph.D. degree in systems science from the Tokyo Institute of Technology, Tokyo, Japan, in 2011. He is currently an Associate Professor with Osaka City University, Osaka, Japan. Previously, he was an Assistant Professor with the University of Tokyo (2013–2014), and a Postdoctoral Fellow with the University of Toronto (2011–2013). His research interests include distributed control of discrete-event systems and cooperative control of networked multi-agent systems. He is the co-author (with W. M. Wonham) of *Supervisory Control of Discrete-Event Systems* (Springer, 2019) and *Supervisor Localization* (Springer, 2016). He is an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL.