Dual-Domain Defenses for Byzantine-Resilient Decentralized Resource Allocation

Runhua Wang, *Member, IEEE*, Qing Ling, *Senior Member, IEEE*, and Zhi Tian, *Fellow, IEEE*

***Abstract*—This paper investigates the problem of decentralized resource allocation in the presence of Byzantine attacks. Such attacks occur when an unknown number of malicious agents send random or carefully crafted messages to their neighbors, aiming to prevent the honest agents from reaching the optimal resource allocation strategy. We characterize these malicious behaviors with the classical Byzantine attacks model, and propose a class of Byzantine-resilient decentralized resource allocation algorithms augmented with dual-domain defenses. The honest agents receive messages containing the (possibly malicious) dual variables from their neighbors at each iteration, and filter these messages with robust aggregation rules. Theoretically, we prove that the proposed algorithms can converge to neighborhoods of the optimal resource allocation strategy, given that the robust aggregation rules are properly designed. Numerical experiments are conducted to corroborate the theoretical results.**

***Index Terms*—Resource allocation, decentralized multi-agent network, Byzantine-resilience.**

I. Introduction

RESOURCE allocation, which aims at assigning limited resources to a group of agents to minimize their costs, is a fundamental problem in network optimization. Existing resource allocation algorithms can be categorized into distributed and decentralized approaches. Distributed resource allocation algorithms rely on a central agent to coordinate all the agents, which often leads to the communication bottleneck at the central agent, and thus results in limited scalability [2], [3]. Consequently, decentralized resource allocation algorithms, which rely on coordination among neighboring agents, have become attractive alternatives. They have been widely applied in various fields, such as smart grids, transportation systems, wireless sensor networks, etc [4], [5], [6]. Solving the decentralized resource allocation problem requires collaboration between neighboring agents. However, such collaboration is not always reliable since some of the agents could be malicious. The aim of this paper is to develop effective decentralized resource allocation algorithms that are resilient to the attacks from the malicious agents.

Decentralized Resource Allocation Algorithms: Existing decentralized resource allocation algorithms can be categorized as continuous-time [7], [8], [9], [10] and discrete-time [11], [12], [13], [14], [15], [16], [17], [18]. In this paper, we focus on discrete-time algorithms. The primary challenge in algorithm design is to satisfy the global resource constraint. Weighted gradient methods have been proposed to guarantee global constraint satisfaction with the aid of feasible initialization [11], [12], [13], but they turn out to be sensitive to perturbations. The work of [11] is based on time-varying networks, while [12] considers fixed networks. The work of [13] utilizes historical information to accelerate the algorithm. On the other hand, primal-dual algorithms handle the global resource constraint via introducing a dual variable [14], [15], [16], [17], [18]. The works of [14], [15] develop decentralized Lagrangian methods, which precisely solve the primal sub-problems while perform a dual gradient step at each iteration. The work of [16] employs the push-pull gradient method to solve the dual problem and proposes a dual gradient tracking algorithm for unbalanced networks. For non-smooth resource allocation problems, decentralized proximal primal-dual algorithms are developed in [17], [18].

The decentralized resource allocation algorithms discussed above perform well when all the agents are honest. However, malicious agents, either spontaneously or by manipulation, are always threats to decentralized networks. These agents do not follow the given algorithmic protocol, but send random or crafted messages to their honest neighbors for the sake of misleading the optimization process. To characterize such behaviors, we use the classical Byzantine attacks model and term the malicious agents as Byzantine agents [19], [20]. We briefly review some general Byzantine-resilient *optimization* algorithms and few Byzantine-resilient *resource allocation* algorithms, as follows.

Byzantine-resilient Algorithms: In the context of distributed optimization, Byzantine-resilient algorithms have been extensively studied. The main idea behind the algorithm design is to use various robust aggregation rules, such as coordinate-wise median [21], Krum [22], [23] and geometric median [24], to

Received 30 October 2023; revised 6 September 2024; accepted 12 October 2024. Date of publication 23 October 2024; date of current version 8 November 2024. The work of Qing Ling was supported in part by NSF China under Grant 62373388, in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2021B1515020094 and Grant 2023B1515040025, in part by the R&D Project of Pazhou Lab (Huangpu) under Grant 2023K0606, and in part by the Guangdong Key Lab of Mathematical Foundations for Artificial Intelligence, The Chinese University of Hong Kong, Shenzhen. The work of Zhi Tian was supported by US NSF under Grant 1939553 and Grant 2231209. An earlier version of this paper was presented in part at the ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) [DOI: 10.1109/ICASSP48485.2024.10446020]. The associate editor coordinating the review of this article and approving it for publication was Dr. Naveen K. D. Venkategowda. *(Corresponding author: Qing Ling.)*

Runhua Wang and Qing Ling are with the School of Computer Science and Engineering and Guangdong Provincial Key Laboratory of Computational Science, Sun Yat-sen University, Guangzhou 510006, China (e-mail: lingqing556@mail.sysu.edu.cn).

Zhi Tian is with the Department of Electrical and Computer Engineering, George Mason University, Fairfax, VA 22030 USA.

This article has supplementary downloadable material available at https://doi.org/10.1109/TSIPN.2024.3485508, provided by the authors.

Digital Object Identifier 10.1109/TSIPN.2024.3485508

filter out malicious messages. However, directly extending this idea into decentralized optimization often cannot guarantee consensus, and thus yields large optimization errors [25].

Given a general Byzantine-resilient decentralized optimization problem, honest agents cooperate to reach a consensual optimal solution that minimizes their average cost function. This is different to the resource allocation problem, where the honest agents are expected to obtain different optimal solutions (namely, allocated resources). Some works focus on deterministic problems [26], [27], [28], [29], [30], [31], [32] and some others consider stochastic problems [25], [33]. Their common feature is to let each honest agent aggregate possibly malicious messages (namely, optimization variables) received from its neighbors in a robust manner.

For Byzantine-resilient decentralized optimization problems with deterministic cost functions, when the optimization variable is a scalar, [26], [27] proposes the trimmed mean (TM) robust aggregation rule, with which each honest agent discards the smallest b and the largest b messages received from its neighbors, followed by averaging the remaining messages and its own. Here b is an estimated upper bound of the number of Byzantine neighbors. A similar approach in [28] lets each honest agent filter b received messages larger and b received messages smaller than its own message, also followed by averaging. For high-dimensional problems, [29], [30] extends TM to coordinate-wise TM (CTM), such that each honest agent performs the TM operation at each dimension. The work of [31] introduces the notion of centerpoint, which is an extension of the robust median aggregation rule to the high-dimensional scenario. In [32], each iteration involves two filtering steps: distance-based and dimension-wise removals. Distance-based removal calculates the Euclidean distances between the received messages and the agent's own message, sorts the distances, and removes b messages with the largest distances. Additionally, messages with extreme values in any dimension are removed.

When the cost functions are stochastic, TM and CTM are also applicable. Besides, the work of [25] proposes iterative outlier scissor (IOS), in which each honest agent iteratively discards b messages that are the farthest from the average of the remaining received messages. The work of [33] proposes self-centered clipping (SCC), in which each honest agent uses its own optimization variable as the center, clips the received messages, and then runs weighted average.

Although the aforementioned Byzantine-resilient decentralized optimization algorithms are proved to be effective, they cannot be directly applied to solve the resource allocation problem. The local optimization variables of the honest agents are coupled with a consensus constraint in the former but with a global resource constraint in the latter. Therefore, in a decentralized resource allocation algorithm, filtering "outliers" from the neighboring optimization variables becomes meaningless. To fill this gap, [3] proposes a primal-dual Byzantine-resilient resource allocation algorithm from a robust optimization perspective, but the proposed algorithm is only applicable in a distributed network with a central server. A Byzantine-resilient decentralized resource allocation (BREDA) algorithm is developed in [34]. In addition to the updates of primal and dual variables, each honest agent maintains an auxiliary variable that dynamically tracks the average of all honest agents' primal variables. Then, CTM is applied to aggregate the neighboring auxiliary variables.

Our Contributions: This paper focuses on the challenging and less-studied Byzantine-resilient decentralized resource allocation problem, and makes the following contributions:

C1) We propose a class of primal-dual Byzantine-resilient decentralized resource allocation algorithms with dual-domain defenses. The key intuition is that the honest agents should reach a consensual dual variable. Therefore, we can let each honest agent filter the received neighboring dual variables with properly designed robust aggregation rules, including but not limited to CTM, IOS and SCC.

C2) Compared with BREDA that defends against Byzantine attacks in the primal domain [34], the proposed algorithms utilize dual-domain defenses, and have the following advantages: (i) maintaining less variables and simpler updates; (ii) allowing more general robust aggregation rules than CTM; (iii) being able to reach dual consensus.

C3) Theoretically, we prove that if the robust aggregation rules are properly designed, the proposed algorithms converge to neighborhoods of the optimal primal-dual pair, and the honest agents are guaranteed to reach consensus in the dual domain even at presence of Byzantine attacks. With numerical experiments, we verify Byzantine-resilience of the proposed algorithms and its advantages over BREDA.

Compared to the short, preliminary conference version [1], this journal version has been significantly extended. We have included comprehensive derivations for the algorithm design, detailed theoretical analysis, as well as additional numerical experiments that deepen the insights presented in [1]. These extensions not only reinforce the theoretical foundation but also enhance the practical relevance of our proposed algorithms.

Paper Organization: This paper is organized as follows. In Section II, we formulate the decentralized resource allocation problem under Byzantine attacks. Section III proposes an attack-free decentralized resource allocation algorithm that operates in the dual domain, and shows its failure under Byzantine attacks. Section IV further proposes a class of Byzantine-resilient decentralized resource allocation algorithms. Section V establishes convergence of the proposed Byzantine-resilient decentralized resource allocation algorithms. Numerical experiments are given in Section VI. Section VII summarizes this paper and discusses future research directions.

Notation: Throughout this paper, $(\cdot)^\top$ stands for the transposition of a vector or a matrix, $\|\cdot\|$ stands for the ℓ_2-norm of a vector or a matrix, $\|\cdot\|_F$ denotes the Frobenius norm of a matrix, and $\langle\cdot,\cdot\rangle$ represents the inner product of vectors. We define $\widetilde{\mathbf{1}} \in \mathbb{R}^J$ and $\mathbf{1} \in \mathbb{R}^H$ as all-one column vectors while $I \in \mathbb{R}^{H\times H}$ as an identity matrix, where J is the number of all agents and H is the number of honest agents.

II. Problem Statement

We consider a decentralized resource allocation problem that involves a network of autonomous agents. The network is modeled as an undirected, connected graph $\widetilde{\mathcal{G}}(\mathcal{J},\widetilde{\mathcal{E}})$ with the set of

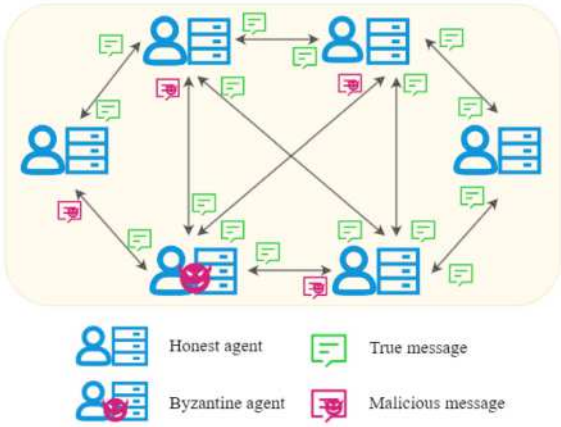

Fig. 1. Decentralized resource allocation under Byzantine attacks.

vertices $\mathcal{J} := \{1, \ldots, J\}$ and the set of edges $\widetilde{\mathcal{E}}$. If $(i, j) \in \widetilde{\mathcal{E}}$, then the two agents i and j are neighbors and can communicate with each other. For agent i, define the set of its neighbors as $\mathcal{N}_i = \{j \mid (i, j) \in \widetilde{\mathcal{E}}\}$. Each agent i possesses a strongly convex local cost function $f_i(\theta_i)$, where $\theta_i \in \mathbb{R}^D$ stands for the amount of local resources and belongs to a compact, convex set C_i. The average amount of local resources, denoted as $\frac{1}{J}\sum_{i\in\mathcal{J}} \theta_i$, equals to a constant vector $s \in \mathbb{R}^D$. When all the agents are honest, the decentralized resource allocation problem is formulated as

$$\begin{aligned} \min_{\widetilde{\Theta}} \quad & \widetilde{f}(\widetilde{\Theta}) = \frac{1}{J}\sum_{i\in\mathcal{J}} f_i(\theta_i), \\ s.t. \quad & \frac{1}{J}\sum_{i\in\mathcal{J}} \theta_i = s, \quad \theta_i \in C_i, \forall i \in \mathcal{J}, \end{aligned} \tag{1}$$

where $\widetilde{\Theta} = [\theta_1, \ldots, \theta_J] \in \mathbb{R}^{JD}$ concatenates all the local variables and $\widetilde{C}$ is the Cartesian product of C_i for all $i \in \mathcal{J}$.

The decentralized resource allocation problem in the form of (1) arises in, for example, economic dispatch in smart grids [35], [36]. The goal is to obtain an optimal generation strategy that minimizes the total generation cost, while satisfying a global power demand constraint and local generator constraints, through cooperation among a network of generators. We will introduce the economic dispatch problem in detail in Section VI, and focus on the case that some of the generators are malicious.

When some of the agents are Byzantine, as shown in Fig. 1, solving (1) is an impossible task, because they will not collaborate with the honest agents during the optimization process. Denote the set of Byzantine agents as $\mathcal{B}$ and the set of honest agents as $\mathcal{H} := \mathcal{J} \setminus \mathcal{B}$. The numbers of Byzantine agents and honest agents are denoted as B and H, respectively. Note that the number and identities of Byzantine agents are not known in advance, but we can roughly estimate an upper bound of the number. For notational convenience, we number the honest agents from 1 to H, and the Byzantine agents from $H+1$ to $H+B$. Consider a subgraph $\mathcal{G}(\mathcal{H}, \mathcal{E})$ of $\widetilde{\mathcal{G}}(\mathcal{J}, \widetilde{\mathcal{E}})$, where $\mathcal{E} = \{(i, j) \in \widetilde{\mathcal{E}}; i, j \in \mathcal{H}\}$ is the set of edges between the honest agents. We assume $\mathcal{G}(\mathcal{H}, \mathcal{E})$ to be connected too so that the honest agents can cooperate. The goal of the honest agents is to solve

$$\begin{aligned} \min_{\Theta} \quad & f(\Theta) := \frac{1}{H}\sum_{i\in\mathcal{H}} f_i(\theta_i), \\ s.t. \quad & \frac{1}{H}\sum_{i\in\mathcal{H}} \theta_i = s, \quad \theta_i \in C_i, \forall i \in \mathcal{H}, \end{aligned} \tag{2}$$

where $\Theta = [\theta_1, \ldots, \theta_H] \in \mathbb{R}^{HD}$ concatenates all the local variables of the honest agents and C is the Cartesian product of C_i for all $i \in \mathcal{H}$.

In (2), we modify the optimization objective to the average cost of the honest agents and consider the average resource constraint of the honest agents. We do not modify the average resource supply s, as the Byzantine agents may still occupy some resources. Adjusting s will inevitably affect the resources allocated to the honest agents.

However, solving (2) is still challenging since the honest agents cannot distinguish their Byzantine neighbors, while the latter can send arbitrarily malicious messages during the optimization process. Therefore, in this paper, we focus on developing Byzantine-resilient decentralized resource allocation algorithms to approximately solve (2).

III. Attack-Free Decentralized Resource Allocation

This section begins with reviewing an attack-free decentralized resource allocation algorithm, which operates in the dual domain, to solve (1).

A. Algorithm Development

The Lagrangian function of (1) is

$$\widetilde{\mathcal{L}}(\widetilde{\Theta}; \widetilde{\lambda}) := \frac{1}{J}\sum_{i\in\mathcal{J}} f_i(\theta_i) + \widetilde{\lambda}^\top \left(\frac{1}{J}\sum_{i\in\mathcal{J}} \theta_i - s\right), \tag{3}$$

where $\widetilde{\lambda} \in \mathbb{R}^D$ is the dual variable. Hence, the dual function $\widetilde{d}(\widetilde{\lambda}) := \min_{\widetilde{\Theta}\in\widetilde{C}} \widetilde{\mathcal{L}}(\widetilde{\Theta}; \widetilde{\lambda})$ is given by

$$\begin{aligned} \widetilde{d}(\widetilde{\lambda}) &:= \min_{\widetilde{\Theta}\in\widetilde{C}} \left\{\frac{1}{J}\sum_{i\in\mathcal{J}} f_i(\theta_i) + \widetilde{\lambda}^\top \left(\frac{1}{J}\sum_{i\in\mathcal{J}} \theta_i - s\right)\right\} \\ &= \frac{1}{J}\sum_{i\in\mathcal{J}} \min_{\theta_i\in C_i} \{f_i(\theta_i) + \widetilde{\lambda}^\top \theta_i\} - \widetilde{\lambda}^\top s \\ &= \frac{1}{J}\sum_{i\in\mathcal{J}} \left(-\max_{\theta_i\in C_i} \{-f_i(\theta_i) - \widetilde{\lambda}^\top \theta_i\}\right) - \widetilde{\lambda}^\top s \\ &= \frac{1}{J}\sum_{i\in\mathcal{J}} -\widetilde{F}_i^*(-\widetilde{\lambda}) - \widetilde{\lambda}^\top s, \end{aligned} \tag{4}$$

where $\widetilde{F}_i^*(\widetilde{\lambda}) := \max_{\theta_i\in C_i} \{\widetilde{\lambda}^\top \theta_i - f_i(\theta_i)\}$. With it, we write the dual problem of (1) as a minimization problem in the form of

$$\min_{\widetilde{\lambda}\in\mathbb{R}^D} \widetilde{g}(\widetilde{\lambda}) = -\widetilde{d}(\widetilde{\lambda}) = \sum_{i\in\mathcal{J}} \widetilde{g}_i(\widetilde{\lambda}), \tag{5}$$

where $\widetilde{g}_i(\widetilde{\lambda}) := \frac{1}{J}\widetilde{F}_i^*(-\widetilde{\lambda}) + \frac{1}{J}\widetilde{\lambda}^\top s$.

Algorithm 1: Attack-Free Decentralized Resource Allocation Algorithm.

Initialization: All agents $i \in \mathcal{J}$ initialize $\boldsymbol{\lambda}_i^0 = \boldsymbol{\lambda}^0$.
for $k = 0, 1, 2, \ldots$ **do**
  for all agents $i \in \mathcal{J}$ **do**
    Compute $\boldsymbol{\theta}_i^k = \arg\min_{\boldsymbol{\theta}_i \in C_i} \{\boldsymbol{\theta}_i^\top \boldsymbol{\lambda}_i^k + f_i(\boldsymbol{\theta}_i)\}$.
    Compute $\boldsymbol{\lambda}_i^{k+\frac{1}{2}} = \boldsymbol{\lambda}_i^k - \gamma^k(\frac{1}{J}s - \frac{1}{J}\boldsymbol{\theta}_i^k)$.
    Broadcast $\boldsymbol{\lambda}_i^{k+\frac{1}{2}}$ to its neighbors.
    Receive $\boldsymbol{\lambda}_j^{k+\frac{1}{2}}$ from its neighbors.
    Aggregate $\boldsymbol{\lambda}_i^{k+1} = \sum_{j \in \mathcal{N}_i \cup \{i\}} \widetilde{e}_{ij} \boldsymbol{\lambda}_j^{k+\frac{1}{2}}$.
  end for
end for

Because $f_i(\cdot)$ is strongly convex, according to the conjugate correspondence theorem in [37], its conjugate function $\widetilde{F}_i^*(\cdot)$ is smooth. By Danskin's theorem [38], the gradient $\nabla \widetilde{F}_i^*(\boldsymbol{\lambda}_i) = \arg\max_{\boldsymbol{\theta}_i \in C_i} \{\boldsymbol{\lambda}_i^\top \boldsymbol{\theta}_i - f_i(\boldsymbol{\theta}_i)\}$. Hence, we have

$$\nabla \widetilde{g}_i(\boldsymbol{\lambda}_i) = \frac{1}{J}s - \frac{1}{J}\arg\min_{\boldsymbol{\theta}_i \in C_i} \{\boldsymbol{\lambda}_i^\top \boldsymbol{\theta}_i + f_i(\boldsymbol{\theta}_i)\}. \tag{6}$$

According to the above discussions, the optimization problem (5) can be solved through decentralized gradient methods [14], [39], [40]. To do so, we let each agent holds a local dual variable $\boldsymbol{\lambda}_i \in \mathbb{R}^D$. The updates of primal and dual variables for all agents $i \in \mathcal{J}$ in the attack-free decentralized resource allocation algorithm at iteration $k+1$ are given by

$$\boldsymbol{\theta}_i^k = \arg\min_{\boldsymbol{\theta}_i \in C_i} \{\boldsymbol{\theta}_i^\top \boldsymbol{\lambda}_i^k + f_i(\boldsymbol{\theta}_i)\}, \tag{7}$$

$$\boldsymbol{\lambda}_i^{k+\frac{1}{2}} = \boldsymbol{\lambda}_i^k - \gamma^k \nabla \widetilde{g}_i(\boldsymbol{\lambda}_i^k) = \boldsymbol{\lambda}_i^k - \gamma^k(\frac{1}{J}s - \frac{1}{J}\boldsymbol{\theta}_i^k), \tag{8}$$

$$\boldsymbol{\lambda}_i^{k+1} = \sum_{j \in \mathcal{N}_i \cup \{i\}} \widetilde{e}_{ij} \boldsymbol{\lambda}_j^{k+\frac{1}{2}}. \tag{9}$$

Therein, $\gamma^k > 0$ is the step size and $\widetilde{e}_{ij} \geq 0$ is the weight assigned by agent i to agent j. Note that $\widetilde{e}_{ij} > 0$ if and only if $(i,j) \in \widetilde{\mathcal{E}}$ or $i = j$. We collect these weights in $\widetilde{E} = [\widetilde{e}_{ij}] \in \mathbb{R}^{J \times J}$, which is assumed to be doubly stochastic. Such an attack-free decentralized resource allocation algorithm is summarized in Algorithm 1.

B. Failure of Attack-Free Decentralized Resource Allocation Algorithm Under Byzantine Attacks

When all the agents are honest, the decentralized resource allocation algorithm outlined in (7)–(9) can effectively solve (1); readers are referred to [14], [39], [40]. However, it fails in the presence of Byzantine attacks. At iteration $k+1$, each honest agent $i \in \mathcal{H}$ updates $\boldsymbol{\lambda}_i^{k+1}$ based on $\boldsymbol{\lambda}_i^{k+\frac{1}{2}}$ from its own and $\boldsymbol{\lambda}_j^{k+\frac{1}{2}}$ from its neighbors $j \in \mathcal{N}_i$. An honest neighbor $j \in \mathcal{N}_i \cap \mathcal{H}$ faithfully sends the message $\boldsymbol{\lambda}_j^{k+\frac{1}{2}}$, but a Byzantine neighbor $j \in \mathcal{N}_i \cap \mathcal{B}$ may send an arbitrarily malicious message $*$ instead of the true message $\boldsymbol{\lambda}_j^{k+\frac{1}{2}}$. We define the message sent by agent j as

$$\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}} = \begin{cases} \boldsymbol{\lambda}_j^{k+\frac{1}{2}}, & j \in \mathcal{H}, \\ *, & j \in \mathcal{B}. \end{cases} \tag{10}$$

The malicious messages sent by the Byzantine agents prevent the honest agents from obtaining the optimal dual variable and corresponding resource allocation strategy. We provide a simple example to illustrate their impact. Assume that the local cost function of agent i is $f_i(\boldsymbol{\theta}_i) = \boldsymbol{\theta}_i^2$, the local resource constraint set is $C_i = [0, 100]$, and the average resource is $s = 50$. The optimal dual variable and resource allocation of agent i are $\boldsymbol{\lambda}_i^* = -100$ and $\boldsymbol{\theta}_i^* = 50$, respectively. According to (7), the update of $\boldsymbol{\theta}_i^{k+1}$ is $\boldsymbol{\theta}_i^{k+1} = \Pi_{[0,100]}(-\frac{\boldsymbol{\lambda}_i^{k+1}}{2})$, the projection of $-\frac{\boldsymbol{\lambda}_i^{k+1}}{2}$ onto [0,100]. A Byzantine agent j can manipulate $\boldsymbol{\lambda}_i^{k+1}$ by (9) to be either 0 or -200 through sending a proper $\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}$. In consequence, honest agent i will obtain resource allocation of either $\boldsymbol{\theta}_i^{k+1} = 0$ or $\boldsymbol{\theta}_i^{k+1} = 100$, which are faraway from the optimal solution.

IV. Byzantine-Resilient Decentralized Resource Allocation

In light of the influence of Byzantine attacks to decentralized resource allocation, we propose a class of Byzantine-resilient decentralized resource allocation algorithms to approximately solve (2) in this section.

A. Algorithm Development

As we have shown in Section III, the decentralized resource allocation algorithm outlined in (7)–(9) fails in the presence of Byzantine attacks. This is due to the vulnerability of the weighted average aggregation in (9) to Byzantine attacks. To address this issue, we replace the weighted average with proper robust aggregation rules, and propose a class of Byzantine-resilient decentralized resource allocation algorithms. The updates of each honest agent $i \in \mathcal{H}$ are given by

$$\boldsymbol{\theta}_i^k = \arg\min_{\boldsymbol{\theta}_i \in C_i} \{\boldsymbol{\theta}_i^\top \boldsymbol{\lambda}_i^k + f_i(\boldsymbol{\theta}_i)\}, \tag{11}$$

$$\boldsymbol{\lambda}_i^{k+\frac{1}{2}} = \boldsymbol{\lambda}_i^k - \gamma^k(\frac{1}{J}s - \frac{1}{J}\boldsymbol{\theta}_i^k), \tag{12}$$

$$\boldsymbol{\lambda}_i^{k+1} = AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}}, \{\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j \in \mathcal{N}_i}), \tag{13}$$

where $AGG_i(\cdot)$ denotes a certain robust aggregation rule of honest agent i. The proposed Byzantine-resilient decentralized resource allocation algorithm is summarized in Algorithm 2.

In this paper, we mainly consider the applications of three well-appreciated robust aggregation rules: CTM, IOS and SCC. Further, we will show that a wide class of robust aggregation rules enable the updates of (11)–(13) to converge to neighborhoods of the optimal resource allocation strategy of (2). The remaining design is to delineate the conditions for "proper" robust aggregation rules.

Robust Aggregation Rules: Intuitively, for an honest agent i, we expect that the output of $AGG_i\left(\boldsymbol{\lambda}_i^{k+\frac{1}{2}}, \{\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j \in \mathcal{N}_i}\right)$

Algorithm 2: Byzantine-Resilient Decentralized Resource Allocation Algorithm.

Initialization: All agents i initialize $\boldsymbol{\lambda}_i^0 = \boldsymbol{\lambda}^0$.
for $k = 0, 1, 2, \ldots$ **do**
  for all honest agents $i \in \mathcal{H}$ **do**
    Compute $\boldsymbol{\theta}_i^k = \arg\min_{\boldsymbol{\theta}_i \in C_i} \{\boldsymbol{\theta}_i^\top \boldsymbol{\lambda}_i^k + f_i(\boldsymbol{\theta}_i)\}$.
    Compute $\boldsymbol{\lambda}_i^{k+\frac{1}{2}} = \boldsymbol{\lambda}_i^k - \gamma^k(\frac{1}{J}s - \frac{1}{J}\boldsymbol{\theta}_i^k)$.
    Broadcast $\boldsymbol{\lambda}_i^{k+\frac{1}{2}}$ to its neighbors.
    Receive $\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}$ from its neighbors.
    Aggregate $\boldsymbol{\lambda}_i^{k+1} = AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}}, \{\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i})$.
  end for
  for all Byzantine agents $i \in \mathcal{B}$ **do**
    Broadcast $\check{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}} = *$ to its neighbors.
  end for
end for

is close to a proper weighted average of the messages from its honest neighbors and its own local dual variable, denoted as $\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}} := \sum_{j\in(\mathcal{N}_i\cap\mathcal{H})\cup\{i\}} e_{ij}\boldsymbol{\lambda}_j^{k+\frac{1}{2}}$ with the weights $\{e_{ij}\}_{j\in\mathcal{H}}$ satisfying $\sum_{j\in(\mathcal{N}_i\cap\mathcal{H})\cup\{i\}} e_{ij} = 1$. We use the maximal value of $\{\|\boldsymbol{\lambda}_j^{k+\frac{1}{2}} - \bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}\|\}_{j\in(\mathcal{N}_i\cap\mathcal{H})\cup\{i\}}$ as the metric to quantify the proximity. Therefore, we follow [25], [41] to characterize a set of robust aggregation rules with a virtual weight matrix and a contraction constant.

Definition 1: Consider a set of robust aggregation rules $\{AGG_i\}_{i\in\mathcal{H}}$. If there exist a constant $\rho \geq 0$ and a matrix $E \in \mathbb{R}^{H\times H}$ whose elements satisfy $e_{ij} \in (0, 1]$ when $j \in (\mathcal{N}_i \cap \mathcal{H}) \cup \{i\}$, $e_{ij} = 0$ when $j \notin (\mathcal{N}_i \cap \mathcal{H}) \cup \{i\}$, and $\sum_{j\in(\mathcal{N}_i\cap\mathcal{H})\cup\{i\}} e_{ij} = 1$ for any $i \in \mathcal{H}$, such that it holds

$$\begin{aligned} &\|AGG_i(\boldsymbol{\lambda}_i, \{\check{\boldsymbol{\lambda}}_j\}_{j\in\mathcal{N}_i}) - \bar{\boldsymbol{\lambda}}_i\| \\ &\leq \rho \max_{j\in(\mathcal{N}_i\cap\mathcal{H})\cup\{i\}} \|\boldsymbol{\lambda}_j - \bar{\boldsymbol{\lambda}}_i\|, \end{aligned} \tag{14}$$

for any $i \in \mathcal{H}$, then ρ is the contraction constant and E is the virtual weight matrix associated with the set of robust aggregation rules $\{AGG_i\}_{i\in\mathcal{H}}$. Here $\bar{\boldsymbol{\lambda}}_i := \sum_{j\in(\mathcal{N}_i\cap\mathcal{H})\cup\{i\}} e_{ij}\boldsymbol{\lambda}_j$.

In the next section, we will prove that if a robust aggregation rule satisfies Definition 1, it is "proper" if the associated ρ is small and E is doubly stochastic.

Remark 1: The work of [41] has demonstrated that CTM, IOS and SCC all satisfy Definition 1 under network conditions stricter than connectedness of the honest agents, and specified their corresponding ρ and E. For example, for each honest agent, CTM requires the number of its honest neighbors to exceed $2b$. Note that the pair of (ρ, E) is not unique. Finding the best pair is beyond the scope of this paper.

There also exist other robust aggregation rules, such as the total-variation-based [42] and attack-detection-based [43] ones, which do not satisfy Definition 1. We will investigate these approaches in our future work.

B. Advantages Over BREDA

Our proposed algorithms have several advantages over BREDA [34]: simplicity, generality and dual consensus. First, at each iteration of BREDA, each honest agent needs to update a primal variable, a dual variable, and an auxiliary variable that tracks the average of the honest primal variables. By contrast, at each iteration of our proposed algorithms, each honest agent only updates two local variables, one is primal and the other is dual. Second, the robust aggregation rule of BREDA is confined to CTM; using other robust aggregation rules lacks convergence guarantee. However, CTM does not fit for the scenario that an honest agent has a large number of Byzantine neighbors, because the number of discarded messages has to be at least twice. This is unfavorable especially when the underlying network is sparse. Instead, our proposed algorithms allow a wide class of robust aggregation rules that satisfy Definition 1. Third, BREDA guarantees the local auxiliary variables to be nearly consensual, but the local dual variables are not necessarily so. We will validate this fact in the numerical experiments. Since the optimal dual variable stands for the shadow price of the resources [44], reaching consensus of the local dual variables is important in various applications. Our proposed algorithms have such a guarantee, as shown in the next section.

V. Convergence Analysis

This section analyzes convergence of the attack-free and Byzantine-resilient decentralized resource allocation algorithms, outlined in (7)–(9) and (11)–(13), respectively.

We begin with several assumptions.

Assumption 1: For any $i \in \mathcal{J}$, the local cost function $f_i(\cdot)$ is u_f-strongly convex and L_f-smooth, and the local constraint set C_i is compact and convex.

Assumption 2: There exist $\widetilde{\boldsymbol{\Theta}}$ and $\boldsymbol{\Theta}$ in the relative interiors of $\widetilde{C}$ and C, such that the constraints $\frac{1}{J}\sum_{i\in\mathcal{J}} \boldsymbol{\theta}_i = s$ and $\frac{1}{H}\sum_{i\in\mathcal{H}} \boldsymbol{\theta}_i = s$ satisfy, respectively.

Assumptions 1 and 2 are common in investigating resource allocation problems, and are satisfied by many applications [3], [35], [36]. With Assumptions 1 and 2, the duality gaps of (1) and (2) are both 0. In addition, the negative dual functions to minimize are also strongly convex and smooth.

Assumption 3: The graphs $\widetilde{\mathcal{G}}(\mathcal{J}, \widetilde{\mathcal{E}})$ and $\mathcal{G}(\mathcal{H}, \mathcal{E})$ are both undirected and connected. The weight matrices $\widetilde{E}$ and E are doubly stochastic and row stochastic, respectively, and satisfy

$$\widetilde{\kappa} := \|\widetilde{E} - \frac{1}{J}\widetilde{\mathbf{1}}\widetilde{\mathbf{1}}^\top\|^2 < 1, \tag{15}$$

$$\kappa := \|E - \frac{1}{H}\mathbf{1}\mathbf{1}^\top E\|^2 < 1. \tag{16}$$

We have emphasized that the connectedness of $\widetilde{\mathcal{G}}$ and $\mathcal{G}$ is necessary. The requirement (15) is common in decentralized optimization. It holds when $\widetilde{e}_{ij} > 0$ if and only if $(i, j) \in \widetilde{\mathcal{E}}$ or $i = j$. The requirement (16) on the associated virtual weight matrix E is in the same form of (15) if E is doubly stochastic, but we allow E to be only row stochastic.

A. Convergence of Attack-Free Decentralized Resource Allocation Algorithm

Denote $(\widetilde{\boldsymbol{\Theta}}^*, \widetilde{\boldsymbol{\lambda}}^*)$ as the optimal primal-dual pair of (1), in which $\widetilde{\boldsymbol{\Theta}}^* \in \mathbb{R}^{JD}$ and $\widetilde{\boldsymbol{\lambda}}^* \in \mathbb{R}^D$. The following theorem shows the convergence of the attack-free decentralized allocation algorithm (7)–(9).

Theorem 1: Consider $\widetilde{\boldsymbol{\Theta}}^{k+1}$ and $\{\boldsymbol{\lambda}_i^{k+1}\}_{i\in\mathcal{J}}$ generated by the attack-free decentralized resource allocation algorithm (7)–(9) and suppose that no Byzantine agents are present. If Assumptions 1–3 hold, then with a proper decreasing step size $\gamma^k = O(\frac{1}{k})$, we have

a) $\lim_{k\to+\infty} \sum_{i\in\mathcal{J}} \|\boldsymbol{\lambda}_i^{k+1} - \widetilde{\boldsymbol{\lambda}}^*\| = 0$,

b) $\lim_{k\to+\infty} \|\widetilde{\boldsymbol{\Theta}}^{k+1} - \widetilde{\boldsymbol{\Theta}}^*\| = 0$.

Theorem 1 shows that the local primal and dual variables generated by (7)–(9) converge to their optima. This matches the classical conclusion for the decentralized gradient method [14], [39], [40]. Those works assume convex and possibly non-smooth cost functions, while we assume strongly convex and smooth cost functions, with which we have performance guarantee for the ensuing Byzantine-resilient algorithms. The proof of Theorem 1 and the conditions on the step size γ^k are in Appendix B of the extended version of this paper [45].

B. Convergence of Byzantine-Resilient Decentralized Resource Allocation Algorithm

Similarly, denote $(\boldsymbol{\Theta}^*, \boldsymbol{\lambda}^*)$ as the optimal primal-dual pair of (2), in which $\boldsymbol{\Theta}^* \in \mathbb{R}^{HD}$ and $\boldsymbol{\lambda}^* \in \mathbb{R}^D$. The following theorem shows the convergence of the Byzantine-resilient decentralized allocation algorithm (11)–(13).

Theorem 2: Consider $\boldsymbol{\Theta}^{k+1}$ and $\{\boldsymbol{\lambda}_i^{k+1}\}_{i\in\mathcal{H}}$ generated by the Byzantine-resilient decentralized resource allocation algorithm (11)–(13). Suppose that Byzantine agents are present but the used robust aggregation rule satisfies (14) in Definition 1. If Assumptions 1–3 hold and the contraction constant ρ satisfies

$$\rho < \frac{1-\kappa}{8\sqrt{H}},$$

then with a proper decreasing step size $\gamma^k = O(\frac{1}{k})$, we have

a) $\limsup_{k\to+\infty} \sum_{i\in\mathcal{H}} \|\boldsymbol{\lambda}_i^{k+1} - \boldsymbol{\lambda}^*\| \le \sqrt{\frac{192\delta^2H^2}{\beta^2}} \cdot \sqrt{1+\frac{9}{\epsilon^3}} \cdot \sqrt{4\rho^2H+\chi^2}$,

b) $\lim_{k\to+\infty} \sum_{i\in\mathcal{H}} \|\boldsymbol{\lambda}_i^{k+1} - \bar{\boldsymbol{\lambda}}^{k+1}\| = 0$,

c) $\limsup_{k\to+\infty} \|\boldsymbol{\Theta}^{k+1} - \boldsymbol{\Theta}^*\| \le \frac{1}{u_f} \cdot \sqrt{\frac{192\delta^2}{\beta^2}} \cdot \sqrt{1+\frac{9}{\epsilon^3}} \cdot \sqrt{4\rho^2H+\chi^2}$,

where $\bar{\boldsymbol{\lambda}}^{k+1} := \frac{1}{H}\sum_{i\in\mathcal{H}} \boldsymbol{\lambda}_i^{k+1}$, $\beta = \frac{1}{H(u_f+L_f)}$, $\epsilon = \kappa - 8\rho\sqrt{H}$, and $\chi^2 := \frac{1}{H}\|E^\top \mathbf{1} - \mathbf{1}\|^2$ quantifies the non-doubly stochasticity of E.

The proof of Theorem 2 and the conditions on the step size γ^k are presented in Appendix A. Theorem 2 demonstrates that if the robust aggregation rule is properly designed such that the associated contraction constant ρ is sufficiently small, then the local primal and dual variables generated by (11)–(13) converge to neighborhoods of their optima. Sizes of the neighborhoods are determined by the associated contraction constant ρ and virtual weight matrix E (more precisely, χ^2). Notably, the local dual variables are guaranteed to reach consensus even under Byzantine attacks.

Compared to the proof of Theorem 1, that of Theorem 2 is more challenging. First, under the Byzantine attacks and with the robust aggregation rule, dual-domain consensus is no longer merited. We discover that ρ must be sufficiently small for reaching consensus. Second, due to the imperfectness during the aggregation, each iteration incurs an error determined by ρ and χ^2. We have to handle such an error within the analysis. Note that when $\rho = 0$ and E is doubly stochastic, Theorem 2 reduces to Theorem 1.

Our analysis is related to but significantly different from that in [25]. The work of [25] considers a general Byzantine-resilient decentralized stochastic non-convex *optimization* problem, and analyzes robust aggregation rules that satisfy Definition 1 in the primal domain. By contrast, we consider a strongly convex *resource allocation* problem, and analyze in the dual domain. The different assumptions lead to different convergence metrics, and the corresponding technical tools are different, too.

Remark 2: Although Algorithm 2 and its convergence analysis in Theorem 2 are only applicable to (1) with an equality constraint, they can be extended to handle the problem with an inequality constraint as well. The extensions can be achieved by incorporating a non-negative projection operation in updating the dual variables of Algorithm 2 and utilizing the non-expansive property of projection in the convergence analysis of Theorem 2.

VI. Numerical Experiments

In this section, we conduct numerical experiments to show the performance of the proposed Byzantine-resilient decen- tralized resource allocation algorithms.

A. Case 1: Synthetic Problem

We first test on a synthetic and scalar case with $D = 1$. Consider a randomly generated network consisting of $J = 100$ agents, where each agent has 15 neighbors. The weight $\widetilde{e}_{ij}$ is set to $\frac{1}{16}$ if and only if $(i,j) \in \widetilde{\mathcal{E}}$ or $i = j$. The total amount of resources is 5000 such that $s = 50$. The local constraint of each agent i is $\boldsymbol{\theta}_i \in C_i = [0, 100]$. Each agent i has a local cost function $f_i(\boldsymbol{\theta}_i) = a_i(\boldsymbol{\theta}_i - b_i)^2$, in which $a_i \sim \mathcal{U}(1,2)$ and $b_i \sim \mathcal{N}(2, 0.6^2)$ with $\mathcal{U}(\cdot,\cdot)$ standing for uniform distribution and $\mathcal{N}(\cdot,\cdot)$ for Gaussian distribution. Such quadratic cost functions is also used in [12], [15], [16].

We randomly select $B = 6$ Byzantine agents by default, but allow each agent to have at most 4 Byzantine neighbors. For the proposed algorithms, we test four types of Byzantine attacks: large-value, small-value, large-value Gaussian, and small-value Gaussian. With large-value attacks, a Byzantine agent sets its message as -0.01. With small-value attacks, a Byzantine agent sets its message as -600. With large-value Gaussian attacks, a Byzantine agent sets its message following a Gaussian distribution with mean -30 and variance 5^2. With small-value Gaussian attacks, a Byzantine agent sets its message following

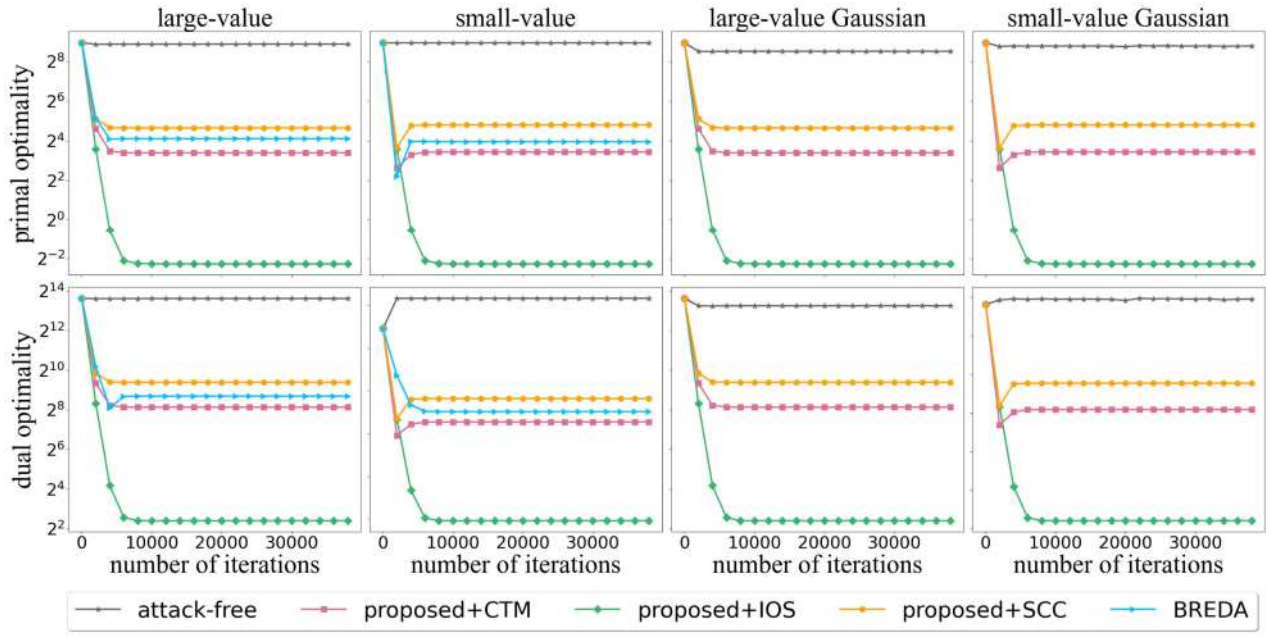

Fig. 2. Primal optimality and dual optimality of the compared algorithms with optimal parameters in Case 1. The number of Byzantine agents is set as 6.

a Gaussian distribution with mean -300 and variance 40^2. We consider three popular robust aggregation rules: CTM, IOS and SCC. The step size is $\gamma^k = (k+1)^{-0.1}$, which is faster than the conservative theoretical step size in the order of $O(\frac{1}{k})$.

We use the attack-free decentralized resource allocation algorithm (7)–(9) as a baseline. Another baseline is BREDA. Note that BREDA defends against Byzantine attacks in the primal domain, whereas our proposed algorithms defend in the dual domain. To enable fair comparisons, for the dual-domain large-value attacks, we generate the corresponding primal-domain attacks such that their effects on the primal variables are almost the same, for our proposed algorithms and BREDA, respectively. Similarly, we also generate the corresponding primal-domain small-value attacks. Thus, with large-value and small-value attacks in BREDA, a Byzantine agent sets its message as 100 and 0, respectively. Note that it is difficult to generate the corresponding primal-domain large-value and small-value Gaussian attacks, and we do not compare with BREDA under these attacks.

Each honest agent sets the parameters b of CTM and IOS to be optimal, as the number of its Byzantine neighbors. In SCC, the clipping threshold τ is determines according to Theorem 3 in [33]. Performance metrics are primal optimality $\|\boldsymbol{\Theta}^k - \boldsymbol{\Theta}^*\|$, dual optimality $\sum_{i\in\mathcal{H}} \|\boldsymbol{\lambda}_i^k - \boldsymbol{\lambda}^*\|$ and dual consensus error $\sum_{i\in\mathcal{H}} \|\boldsymbol{\lambda}_i^k - \bar{\boldsymbol{\lambda}}^k\|^2$. In the extended version of this paper [45], we also consider the setting of non-optimal parameters, as well as evaluate on more performance metrics, including cost optimality $\|f(\boldsymbol{\Theta}^k) - f(\boldsymbol{\Theta}^*)\|$ and constraint violation $\|\frac{1}{H}\sum_{i\in\mathcal{H}} \boldsymbol{\theta}_i^k - s\|$. Besides, we test the sensitivity to different number of Byzantine agents, too.

Fig. 2 illustrates that the attack-free decentralized resource allocation algorithm (7)–(9) fails under all Byzantine attacks. By contrast, the proposed algorithms and BREDA demonstrate satisfactory Byzantine-resilience. Among the robust aggregation rules used in our proposed algorithms, IOS performs the best and CTM is better than SCC in terms of primal optimality and dual optimality. To see the reason, recall that Theorem 2 shows the primal optimality and dual optimality are both in the order of $O(\rho^2 + \chi^2)$. We calculate the corresponding bounds of $\rho^2 + \chi^2$ in Table I according to Lemmas 3–5 in [41]. From the smallest to the largest are respectively IOS, CTM and SCC, which validates our theoretical findings. In the numerical experiments, we observe that even though the contraction factor ρ of SCC is greater than 1, the primal-dual optimality error of SCC converges to a fixed value other than explodes. However, in the theoretical analysis, we require $\rho < \frac{1-\kappa}{8\sqrt{H}} < 1$ to ensure convergence. This stricter requirement arises because we must guarantee consensus of the dual variables under Byzantine attacks. The discrepancy between our experimental and theoretical results is understandable, since the theoretical analysis must account for the worst-case scenarios and thus require more conservative conditions.

TABLE I
BOUNDS OF ρ^2 AND χ^2 FOR CASE 1

	ρ^2	χ^2	$\rho^2+\chi^2$
CTM	0.44	0.0031	0.44
IOS	0.11	0	0.11
SCC	2.75	0	2.75

Fig. 2 also reveals that BREDA is worse than the proposed algorithms with proper robust aggregation rules. To further highlight the advantages of our proposed algorithms, we list the dual consensus errors in Table II. No matter the types of Byzantine attacks and robust aggregation rules, the proposed algorithms achieve nearly perfect dual consensus. By contrast, BREDA cannot guarantee dual consensus. This phenomenon shows the benefits of the dual-domain defenses.

TABLE II
DUAL CONSENSUS ERRORS WITH OPTIMAL PARAMETERS FOR CASE 1

	large-value	small-value	large-value Gaussian	small-value Gaussian
BREDA	105.70	121.09	/	/
proposed+CTM	1.20e-02	1.07e-02	1.20e-02	1.07e-02
proposed+IOS	1.09e-02	1.09e-02	1.09e-02	1.09e-02
proposed+SCC	3.36e-02	3.16e-02	3.36e-02	3.16e-02

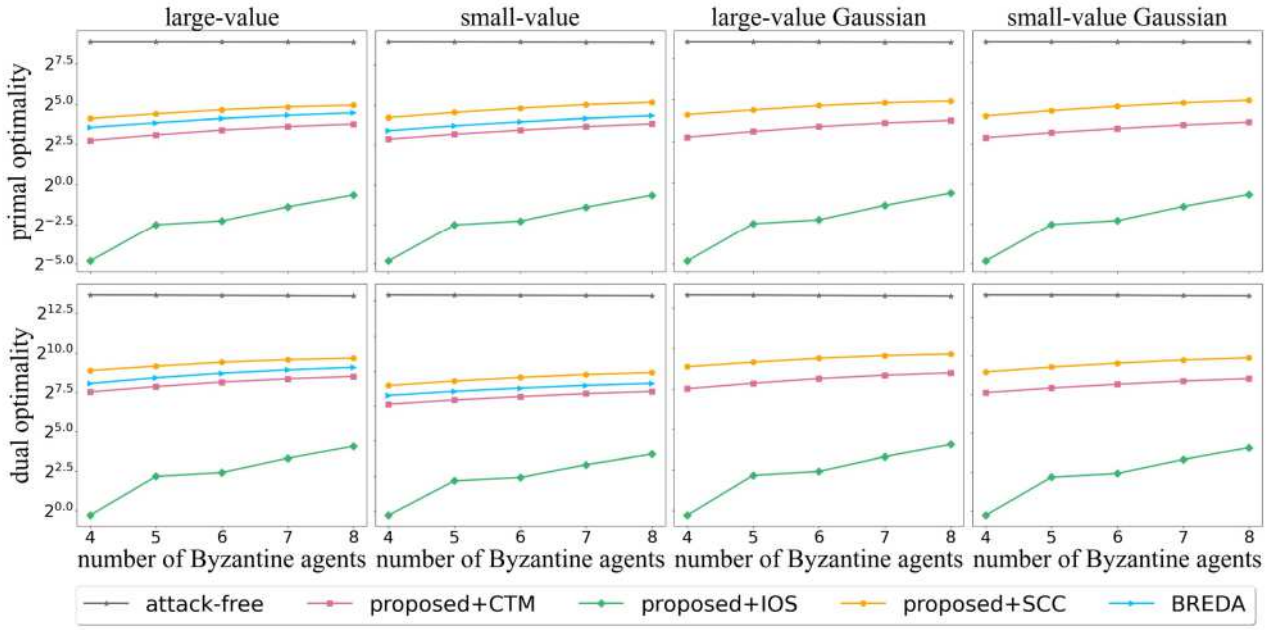

Fig. 3. Primal optimality and dual optimality of the compared algorithms with optimal parameters in Case 1. The number of Byzantine agents is set as 4, 5, 6, 7, and 8.

In Fig. 3, we check the sensitivity of the compared algorithms to the number of Byzantine agents B by setting B as 4, 5, 6, 7 and 8. The attack-free decentralized resource allocation algorithm (7)–(9) fails for any value of B. By contrast, both BREDA and our proposed algorithms demonstrate satisfactory resilience, and their performance is steady when B varies.

B. Case 2: Economic Dispatch for IEEE 118-Bus Test System

We next consider a power dispatch problem for the IEEE 118-bus test system, which contains 54 generators [46]. Each generator i has a local power θ_i and a corresponding cost function $f_i(\theta_i) = \eta_i\theta_i^2 + \zeta_i\theta_i + \xi_i$, where $\eta_i \in [0.0024, 0.0697]$, $\zeta_i \in [8.3391, 37.6968]$, and $\xi_i \in [6.78, 74.33]$. The local constraint of each agent i is $\theta_i \in [\theta_i^{\min}, \theta_i^{\max}]$, where $\theta_i^{\min} \in [5, 150]$ and $\theta_i^{\max} \in [30, 420]$. The total amount of resources is set as 6000, such that $s = \frac{6000}{54}$ [14]. To test the performance of the proposed algorithms, we randomly select one Byzantine agent out of the 54 generators and apply different types of Byzantine attacks, including large-value, small-value, large-value Gaussian, and small-value Gaussian. For large-value attacks, the Byzantine generator sets its message as -0.01, whereas for small-value attacks, the Byzantine generator sets its message as -100. For large-value Gaussian attacks, the Byzantine generator sets its message following a Gaussian distribution with mean -10 and variance 5^2. For small-value Gaussian attacks, the Byzantine generator sets its message following a Gaussian distribution with mean -50 and variance 10^2. We also design the corresponding larger-value and smaller-value attacks for BREDA, where the Byzantine generator sets its message as 420 and 5, respectively. The weight matrix $\widetilde{E}$ is constructed according to the Metropolis constant weight rule [47]. The parameters b and τ are optimal. The step size for the proposed algorithms is determined as $\gamma^k = (k+1)^{-0.7}$.

TABLE III
BOUNDS OF ρ^2 AND χ^2 FOR CASE 2

	ρ^2	χ^2	$\rho^2 + \chi^2$
CTM	0.024	0.11	0.134
IOS	0.006	0	0.006
SCC	0.965	0	0.965

Fig. 4 demonstrates the failure of the attack-free decentralized resource allocation algorithm, as well as the resilience of the proposed algorithms and BREDA against various Byzantine attacks. We also calculate the corresponding bounds of $\rho^2 + \chi^2$ of the robust aggregation rules IOS, CTM, and SCC, as presented in Table III. Observe that a smaller bound of $\rho^2 + \chi^2$ leads to better performance, which has been predicted by our theoretical findings.

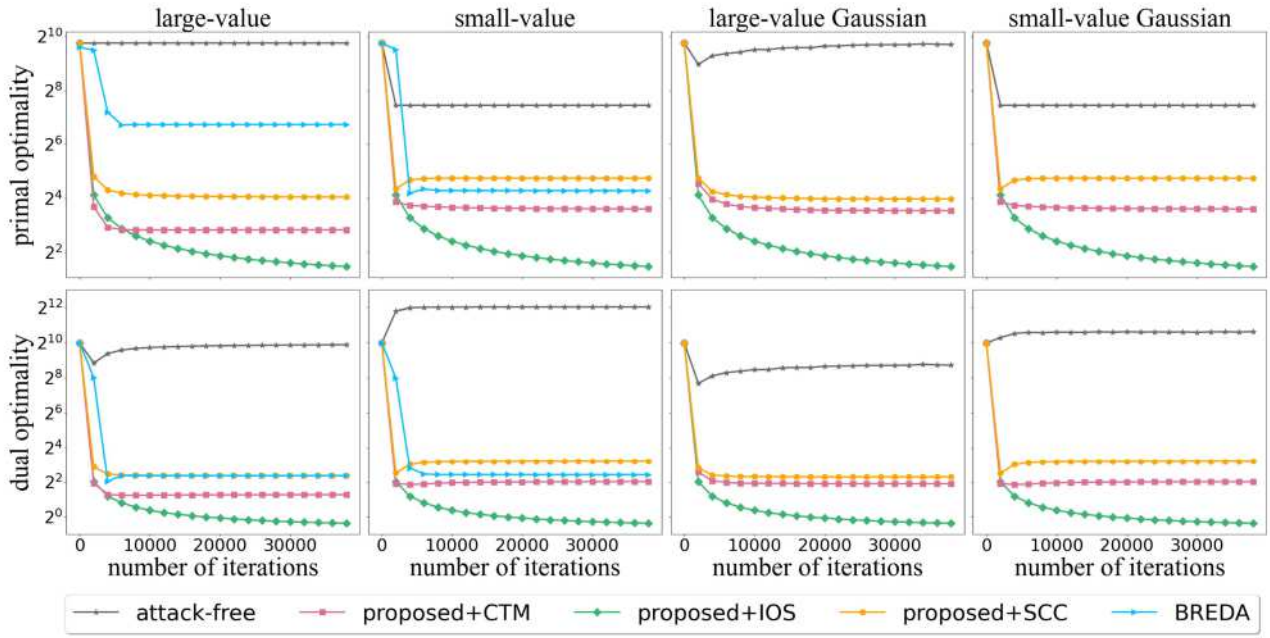

Fig. 4. Primal optimality and dual optimality of the compared algorithms with optimal parameters in Case 2. The number of Byzantine agents is set as 6.

TABLE IV
DUAL CONSENSUS ERRORS WITH OPTIMAL PARAMETERS FOR CASE 2

	large-value	small-value	large-value Gaussian	small-value Gaussian
BREDA	0.51	0.49	/	/
proposed+CTM	2.16e-04	3.28e-03	3.48e-03	3.28e-03
proposed+IOS	3.37e-03	3.37e-03	3.37e-03	3.37e-03
proposed+SCC	3.55e-03	3.23e-03	3.54e-03	3.23e-03

According to Fig. 4, BREDA performs worse than the proposed algorithms with proper robust aggregation rules. We calculate the dual consensus errors of the proposed algorithms with different robust aggregation rules and BREDA, as presented in Table IV. The proposed algorithms achieve nearly consensual dual variables and BREDA does not.

VII. Conclusions and Future Work

In this paper, we address the challenging Byzantine-resilience issue in decentralized resource allocation. We propose a class of Byzantine-resilient algorithms that leverage robust aggregation rules within a dual-domain defense framework. Given that the robust aggregation rules are properly designed, we prove that the primal and dual variables of the honest agents converge to the neighborhoods of their optima, while the dual variables are able to reach consensus. This dual-domain defense approach not only simplifies the algorithmic updates but also enhances the overall Byzantine-resilience. Our numerical experiments further demonstrate the resilience of the proposed algorithms against various Byzantine attacks, confirming their practical utility.

In the future, we plan to extend our algorithm development and theoretical analysis to stochastic and online decentralized resource allocation problems under Byzantine attacks, which are of particular importance for time-sensitive applications.

Appendix A
Proof of Theorem 2

A1 Part a of Theorem 2

According to the update of $\boldsymbol{\lambda}_i^{k+1}$ in Algorithm 2, we have

$$
\begin{aligned}
&\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2\\
&=\|\frac{1}{H}\sum_{i\in\mathcal{H}}AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}},\{\tilde{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i})-\boldsymbol{\lambda}^*\|^2\\
&=\|\frac{1}{H}\sum_{i\in\mathcal{H}}AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}},\{\tilde{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i})-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}+\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}-\boldsymbol{\lambda}^*\|^2\\
&\le\frac{1}{v_1}\|\frac{1}{H}\sum_{i\in\mathcal{H}}AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}},\{\tilde{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i})-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2\\
&\quad+\frac{1}{1-v_1}\|\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}-\boldsymbol{\lambda}^*\|^2\\
&=\frac{1}{v_1}\|\frac{1}{H}\sum_{i\in\mathcal{H}}AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}},\{\tilde{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i})-\frac{1}{H}\sum_{i\in\mathcal{H}}\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}\\
&\quad+\frac{1}{H}\sum_{i\in\mathcal{H}}\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2+\frac{1}{1-v_1}\|\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}-\boldsymbol{\lambda}^*\|^2\\
&\le\frac{2}{v_1}\|\frac{1}{H}\sum_{i\in\mathcal{H}}AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}},\tilde{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}(j\in\mathcal{N}_i))-\frac{1}{H}\sum_{i\in\mathcal{H}}\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}\|^2
\end{aligned}
$$

$$+\frac{2}{v_1}\|\frac{1}{H}\sum_{i\in\mathcal{H}}\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2+\frac{1}{1-v_1}\|\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}-\boldsymbol{\lambda}^*\|^2$$
$$\leq \underbrace{\frac{2}{v_1 H}\sum_{i\in\mathcal{H}}\|AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}},\{\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i})-\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}\|^2}_{T_1}$$
$$+\underbrace{\frac{2}{v_1}\|\frac{1}{H}\sum_{i\in\mathcal{H}}\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2}_{T_2}+\underbrace{\frac{1}{1-v_1}\|\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}-\boldsymbol{\lambda}^*\|^2}_{T_3}, \tag{17}$$

where v_1 is any positive constant in (0,1). To derive the first inequality, we use $\|a+b\|^2\leq\frac{1}{v}\|a\|^2+\frac{1}{1-v}\|b\|^2$ for any positive constant $v\in(0,1)$. The last inequality holds because $(a_1+\cdots+a_H)^2\leq H(a_1^2+\cdots+a_H^2)$. Next, we analyze T_1, T_2 and T_3 in turn.

Bounding T_1: According to (14) in Definition 1, T_1 can be bounded by

$$\begin{aligned}
T_1&\leq\frac{2}{v_1 H}\sum_{i\in\mathcal{H}}\rho^2\max_{j\in\mathcal{N}_i\cap\mathcal{H}\cup\{i\}}\|\boldsymbol{\lambda}_j^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}\|^2\\
&=\frac{2\rho^2}{v_1 H}\sum_{i\in\mathcal{H}}\max_{j\in\mathcal{N}_i\cap\mathcal{H}\cup\{i\}}\|\boldsymbol{\lambda}_j^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}+\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}\|^2\\
&\leq\frac{4\rho^2}{v_1 H}\sum_{i\in\mathcal{H}}\max_{j\in\mathcal{N}_i\cap\mathcal{H}\cup\{i\}}\|\boldsymbol{\lambda}_j^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2\\
&\quad+\frac{4\rho^2}{v_1 H}\sum_{i\in\mathcal{H}}\|\bar{\boldsymbol{\lambda}}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2\\
&\leq\frac{4\rho^2}{v_1 H}\sum_{i\in\mathcal{H}}\max_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2\\
&\quad+\frac{4\rho^2}{v_1 H}\sum_{i\in\mathcal{H}}\max_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2\\
&=\frac{8\rho^2}{v_1}\max_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2.
\end{aligned}\tag{18}$$

Define $\Lambda=[\cdots,\boldsymbol{\lambda}_i,\cdots]\in\mathbb{R}^{H\times D}$ that collects $\boldsymbol{\lambda}_i$ of all honest agents $i\in\mathcal{H}$. Combining the fact $\max_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2\leq\|\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\|_F^2$ and (18), we obatin

$$T_1\leq\frac{8\rho^2}{v_1}\|\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\|_F^2. \tag{19}$$

Bounding T_2: By the definition of $\bar{\boldsymbol{\lambda}}_i=\sum_{j\in\mathcal{N}_i\cap\mathcal{H}\cup\{i\}}e_{i,j}\boldsymbol{\lambda}_j$, we have

$$\begin{aligned}
T_2&=\frac{2}{v_1}\|\frac{1}{H}\sum_{i\in\mathcal{H}}\sum_{j\in\mathcal{N}_i\cap\mathcal{H}\cup\{i\}}e_{i,j}\boldsymbol{\lambda}_j^{k+\frac{1}{2}}-\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}\|^2\\
&=\frac{2}{v_1}\|\frac{1}{H}\mathbf{1}^\top E\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\|^2\\
&=\frac{2}{v_1}\|\frac{1}{H}\mathbf{1}^\top\left(E\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\right)\|^2\\
&=\frac{2}{v_1 H^2}\|\mathbf{1}^\top(E\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}})\|^2\\
&=\frac{2}{v_1 H^2}\|(\mathbf{1}^\top E-\mathbf{1}^\top)(\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}})\|^2\\
&\leq\frac{2}{v_1 H^2}\|E^\top\mathbf{1}-\mathbf{1}\|^2\|\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\|^2.
\end{aligned}\tag{20}$$

To drive the last equality, we use Definition 1 that the virtual weight matrix E is row stochastic.

Define $\chi^2=\frac{1}{H}\|E^\top\mathbf{1}-\mathbf{1}\|^2$ to quantify how non-column stochastic the virtual weight matrix E is. Applying the fact $\|\cdot\|^2\leq\|\cdot\|_F^2$ to the right-hand side of (20), we have

$$T_2\leq\frac{2\chi^2}{v_1 H}\|\Lambda^{k+\frac{1}{2}}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\|_F^2. \tag{21}$$

Bounding T_3: Averaging both sides of (12) over $i\in\mathcal{H}$, we have

$$\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}=\bar{\boldsymbol{\lambda}}^k-\frac{\gamma^k}{H}\sum_{i\in\mathcal{H}}(\frac{1}{J}\boldsymbol{s}-\frac{1}{J}\boldsymbol{\theta}_i^k). \tag{22}$$

The dual problem of (2) can be written as a minimization problem in the form of

$$\min_{\boldsymbol{\lambda}\in\mathbb{R}^D}g(\boldsymbol{\lambda})=\sum_{i\in\mathcal{H}}g_i(\boldsymbol{\lambda}), \tag{23}$$

where $g_i(\boldsymbol{\lambda}):=\frac{1}{H}F_i^*(-\boldsymbol{\lambda})+\frac{1}{H}\boldsymbol{\lambda}^\top\boldsymbol{s}$ and $F_i^*(\boldsymbol{\lambda}):=\max_{\boldsymbol{\theta}_i\in C_i}\{\boldsymbol{\lambda}^\top\boldsymbol{\theta}_i-f_i(\boldsymbol{\theta}_i)\}$. Based on the definition of $g_i(\boldsymbol{\lambda}):=\frac{1}{H}F_i^*(-\boldsymbol{\lambda})+\frac{1}{H}\boldsymbol{\lambda}^\top\boldsymbol{s}$ and Danskin's theorem [38], we have

$$\nabla g_i(\boldsymbol{\lambda}_i)=\frac{1}{H}\boldsymbol{s}-\frac{1}{H}\arg\min_{\boldsymbol{\theta}_i\in C_i}\{\boldsymbol{\lambda}_i^\top\boldsymbol{\theta}_i+f_i(\boldsymbol{\theta}_i)\}. \tag{24}$$

Combining (11), (22) and (24), we can obtain

$$\bar{\boldsymbol{\lambda}}^{k+\frac{1}{2}}=\bar{\boldsymbol{\lambda}}^k-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\boldsymbol{\lambda}_i^k). \tag{25}$$

Substituting (25) into T_3, we have

$$\begin{aligned}
T_3&=\frac{1}{1-v_1}\|\bar{\boldsymbol{\lambda}}^k-\boldsymbol{\lambda}^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\boldsymbol{\lambda}}^k)\\
&\quad+\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\boldsymbol{\lambda}}^k)-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\boldsymbol{\lambda}_i^k)\|^2\\
&=\frac{1}{1-v_1}\|\bar{\boldsymbol{\lambda}}^k-\boldsymbol{\lambda}^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\boldsymbol{\lambda}}^k)\|^2\\
&\quad+\frac{(\gamma^k)^2}{1-v_1}\|\frac{1}{J}\sum_{i\in\mathcal{H}}(\nabla g_i(\bar{\boldsymbol{\lambda}}^k)-\nabla g_i(\boldsymbol{\lambda}_i^k))\|^2+\frac{2\gamma^k}{1-v_1}\cdot\\
&\quad\left\langle\bar{\boldsymbol{\lambda}}^k-\boldsymbol{\lambda}^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\boldsymbol{\lambda}}^k),\frac{1}{J}\sum_{i\in\mathcal{H}}(\nabla g_i(\bar{\boldsymbol{\lambda}}^k)\right.\\
&\quad\left.-\nabla g_i(\boldsymbol{\lambda}_i^k))\right\rangle\leq\frac{1}{1-v_1}\|\bar{\boldsymbol{\lambda}}^k-\boldsymbol{\lambda}^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\boldsymbol{\lambda}}^k)\|^2\\
&\quad+\frac{(\gamma^k)^2}{1-v_1}\|\frac{1}{J}\sum_{i\in\mathcal{H}}(\nabla g_i(\bar{\boldsymbol{\lambda}}^k)-\nabla g_i(\boldsymbol{\lambda}_i^k))\|^2
\end{aligned}$$

$$+\frac{v_2^{-1}\gamma^k}{1-v_1}\|\frac{1}{J}\sum_{i\in\mathcal{H}}(\nabla g_i(\bar{\lambda}^k)-\nabla g_i(\lambda_i^k))\|^2$$
$$+\frac{v_2\gamma^k}{1-v_1}\|\bar{\lambda}^k-\lambda^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)\|^2$$
$$\le\frac{1+v_2\gamma^k}{1-v_1}\|\bar{\lambda}^k-\lambda^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)\|^2$$
$$+\frac{\gamma^k(\gamma^k+v_2^{-1})H}{(1-v_1)J^2}\sum_{i\in\mathcal{H}}\|\nabla g_i(\bar{\lambda}^k)-\nabla g_i(\lambda_i^k)\|^2, \quad (26)$$

where $v_2>0$ is any positive constant. To drive the first inequality, we use $2a^\top b\le v^{-1}\|a\|^2+v\|b\|^2$ for any $v>0$. The last inequality holds because $(a_1+\cdots+a_H)^2\le H(a_1^2+\cdots+a_H^2)$. Next, we analyze the first and second terms at the right-hand side of (26) in turn.

According to the fact that $\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)=0$, we have

$$\frac{1+v_2\gamma^k}{1-v_1}\|\bar{\lambda}^k-\lambda^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)\|^2$$
$$=\frac{1+v_2\gamma^k}{1-v_1}\|\bar{\lambda}^k-\lambda^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)\|^2$$
$$=\frac{(1+v_2\gamma^k)(\gamma^k)^2H^2}{(1-v_1)J^2}\|\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)-\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)\|^2$$
$$+\frac{1+v_2\gamma^k}{1-v_1}\|\bar{\lambda}^k-\lambda^*\|^2-\frac{2\gamma^k(1+v_2\gamma^k)H}{(1-v_1)J}\cdot$$
$$\left\langle\bar{\lambda}^k-\lambda^*,\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)-\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)\right\rangle. \quad (27)$$

For $\left\langle\bar{\lambda}^k-\lambda^*,\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)-\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)\right\rangle$, the last term at the right-hand side of (27), we have the following bound. According to Lemma 2, $g_i(\cdot)$ is $\frac{1}{HL_f}$-strongly convex and $\frac{1}{Hu_f}$-smooth. By Lemma 3 in [48], since $\frac{1}{H}\sum_{i\in\mathcal{H}}g_i(\cdot)$ is $\frac{1}{HL_f}$-strongly convex and $\frac{1}{Hu_f}$-smooth, we have

$$\left\langle\bar{\lambda}^k-\lambda^*,\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)-\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)\right\rangle$$
$$\ge\alpha\|\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)-\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)\|^2+\beta\|\bar{\lambda}^k-\lambda^*\|^2, \quad (28)$$

where $\alpha=\frac{Hu_fL_f}{u_f+L_f}$ and $\beta=\frac{1}{H(u_f+L_f)}$. Substituting (28) into (27) and rearranging the terms, we have

$$\frac{1+v_2\gamma^k}{1-v_1}\|\bar{\lambda}^k-\lambda^*-\frac{\gamma^k}{J}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)\|^2$$
$$\le\frac{(1+v_2\gamma^k)(1-2\gamma^k\beta\cdot\frac{H}{J})}{1-v_1}\|\bar{\lambda}^k-\lambda^*\|^2$$
$$+\frac{(1+v_2\gamma^k)((\gamma^k)^2\cdot\frac{H}{J}-2\gamma^k\alpha)}{1-v_1}\cdot$$
$$\frac{H}{J}\|\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\bar{\lambda}^k)-\frac{1}{H}\sum_{i\in\mathcal{H}}\nabla g_i(\lambda^*)\|^2$$
$$\le\frac{(1+v_2\gamma^k)(1-\gamma^k\beta\cdot\frac{H}{J})}{1-v_1}\|\bar{\lambda}^k-\lambda^*\|^2, \quad (29)$$

where the last inequality holds with a proper step size γ^k satisfying $(\gamma^k)^2\cdot\frac{H}{J}-2\gamma^k\alpha\le0$.

Since $g_i(\cdot)$ is $\frac{1}{Hu_f}$-smooth, we obtain

$$\frac{\gamma^k(\gamma^k+v_2^{-1})}{(1-v_1)H}\cdot\frac{H^2}{J^2}\sum_{i\in\mathcal{H}}\|\nabla g_i(\bar{\lambda}^k)-\nabla g_i(\lambda_i^k)\|^2$$
$$\le\frac{\gamma^k(\gamma^k+v_2^{-1})}{(1-v_1)H^3u_f^2}\cdot\frac{H^2}{J^2}\sum_{i\in\mathcal{H}}\|\lambda_i^k-\bar{\lambda}^k\|^2$$
$$=\frac{\gamma^k(\gamma^k+v_2^{-1})}{(1-v_1)H^3u_f^2}\cdot\frac{H^2}{J^2}\|\Lambda^k-\frac{1}{H}11^\top\Lambda^k\|_F^2. \quad (30)$$

Substituting (29) and (30) into (26) and rearranging the terms, we obtain

$$T_3\le\frac{(1+v_2\gamma^k)(1-\gamma^k\beta\cdot\frac{H}{J})}{1-v_1}\|\bar{\lambda}^k-\lambda^*\|^2$$
$$+\frac{\gamma^k(\gamma^k+v_2^{-1})}{(1-v_1)H^3u_f^2}\cdot\frac{H^2}{J^2}\|\Lambda^k-\frac{1}{H}11^\top\Lambda^k\|_F^2. \quad (31)$$

Substituting (19), (21) and (31) into (17) and rearranging the terms, we have

$$\|\bar{\lambda}^{k+1}-\lambda^*\|^2\le\frac{(1+v_2\gamma^k)(1-\gamma^k\beta\cdot\frac{H}{J})}{1-v_1}\|\bar{\lambda}^k-\lambda^*\|^2$$
$$+\frac{\gamma^k(\gamma^k+v_2^{-1})}{(1-v_1)H^3u_f^2}\cdot\frac{H^2}{J^2}\|\Lambda^k-\frac{1}{H}11^\top\Lambda^k\|_F^2$$
$$+\frac{8\rho^2}{v_1}\|\Lambda^{k+\frac{1}{2}}-\frac{1}{H}11^\top\Lambda^{k+\frac{1}{2}}\|_F^2$$
$$+\frac{2\chi^2}{v_1H}\|\Lambda^{k+\frac{1}{2}}-\frac{1}{H}11^\top\Lambda^{k+\frac{1}{2}}\|_F^2. \quad (32)$$

Setting $v_1=\frac{\gamma^k\beta\cdot\frac{H}{J}}{4}\in(0,1)$ and $v_2=\frac{\beta\cdot\frac{H}{J}}{2(1-\gamma^k\beta\cdot\frac{H}{J})}>0$, from (32) we have

$$\|\bar{\lambda}^{k+1}-\lambda^*\|^2$$
$$\le(1-\frac{\gamma^k\beta\cdot\frac{H}{J}}{4})\|\bar{\lambda}^k-\lambda^*\|^2+\frac{4\gamma^k}{\beta H^2u_f^2J}\|\Lambda^k-\frac{1}{H}11^\top\Lambda^k\|_F^2$$
$$+\frac{8(4\rho^2H+\chi^2)J}{\gamma^k\beta H^2}\|\Lambda^{k+\frac{1}{2}}-\frac{1}{H}11^\top\Lambda^{k+\frac{1}{2}}\|_F^2. \quad (33)$$

Substituting (64) in Lemma 3 into (33) and rearranging the terms, we obtain

$$\|\bar{\lambda}^{k+1}-\lambda^*\|^2$$
$$\le\left(1-\frac{\gamma^k\beta\cdot\frac{H}{J}}{4}\right)\|\bar{\lambda}^k-\lambda^*\|^2+\frac{4\gamma^k}{\beta H^2u_f^2J}\|\Lambda^k-\frac{1}{H}11^\top\Lambda^k\|_F^2$$
$$+\frac{24(4\rho^2H+\chi^2)J}{\gamma^k\beta H^2}\|\Lambda^k-\frac{1}{H}11^\top\Lambda^k\|_F^2$$

$$+\frac{48\gamma^k\delta^2(4\rho^2H+\chi^2)H}{\beta J}. \tag{34}$$

Based on Lemma 4, we can rewrite (34) as

$$\begin{aligned}&\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2\\ &\le\left(1-\frac{\gamma^k\beta\cdot\frac{H}{J}}{4}\right)\|\bar{\boldsymbol{\lambda}}^k-\boldsymbol{\lambda}^*\|^2+\frac{72(\gamma^k)^3\delta^2H}{\beta\epsilon^3u_f^2J^3}\\ &\quad+\frac{432\gamma^k(4\rho^2H+\chi^2)\delta^2H}{\beta\epsilon^3J}+\frac{48\gamma^k\delta^2(4\rho^2H+\chi^2)H}{\beta J}.\end{aligned} \tag{35}$$

Set a proper decreasing step size $\gamma^k=\frac{4}{\beta\cdot\frac{H}{J}(k_0+k)}$, where $k_0>1$ is a any positive integer. Thus, $1-\frac{\gamma^k\beta\cdot\frac{H}{J}}{4}=1-\frac{1}{k_0+k}$ and (35) can be rewritten as

$$\begin{aligned}&\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2\\ &\le\left(1-\frac{1}{k_0+k}\right)\|\bar{\boldsymbol{\lambda}}^k-\boldsymbol{\lambda}^*\|^2+\frac{1}{(k_0+k)^3}\cdot\frac{4608\delta^2}{\beta^4\epsilon^3u_f^2H^2}\\ &\quad+\frac{1}{k_0+k}\cdot\frac{192\delta^2}{\beta^2}\cdot\left(1+\frac{9}{\epsilon^3}\right)\cdot(4\rho^2H+\chi^2).\end{aligned} \tag{36}$$

Then we rewrite (36) recursively and obtain

$$\begin{aligned}&\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2\\ &\le\underbrace{\prod_{k'=0}^{k}\left(1-\frac{1}{k_0+k-k'}\right)}_{T_4}\|\bar{\boldsymbol{\lambda}}^0-\boldsymbol{\lambda}^*\|^2\\ &+[\prod_{k'=0}^{k-1}\left(1-\frac{1}{k_0+k-k'}\right)\frac{1}{(k_0+0)^3}+\cdots+\\ &\underbrace{\prod_{k'=0}^{0}\frac{(1-\frac{1}{k_0+k-k'})}{(k_0+k-1)^3}+\frac{1}{(k_0+k)^3}]}_{T_5}\cdot\frac{4608\delta^2}{\beta^4\epsilon^3u_f^2H^2}\\ &+[\prod_{k'=0}^{k-1}\left(1-\frac{1}{k_0+k-k'}\right)\frac{1}{k_0+0}+\cdots+\\ &\underbrace{\prod_{k'=0}^{0}\frac{(1-\frac{1}{k_0+k-k'})}{k_0+k-1}+\frac{1}{k_0+k}]}_{T_6}\cdot\frac{192\delta^2(4\rho^2H+\chi^2)(1+\frac{9}{\epsilon^3})}{\beta^2}.\end{aligned} \tag{37}$$

Next, we analyze T_4, T_5 and T_6 in turn.

For T_4, we have

$$\begin{aligned}T_4&=\frac{k_0+k-1}{k_0+k}\cdot\frac{k_0+k-2}{k_0+k-1}\cdot\cdots\cdot\frac{k_0-1}{k_0}\\ &=\frac{k_0-1}{k_0+k}.\end{aligned} \tag{38}$$

For T_5, we have

$$\begin{aligned}T_5&=\frac{k_0}{k_0+k}\cdot\frac{1}{(k_0+0)^3}+\frac{k_0+1}{k_0+k}\cdot\frac{1}{(k_0+1)^3}+\cdots\\ &\quad+\frac{k_0+k-1}{k_0+k}\cdot\frac{1}{(k_0+k-1)^3}+\frac{1}{(k_0+k)^3}\\ &=\frac{1}{k_0+k}\left[\frac{1}{(k_0)^2}+\frac{1}{(k_0+1)^2}+\cdots\right.\\ &\quad\left.+\frac{1}{(k_0+k-1)^2}+\frac{1}{(k_0+k)^2}\right]\\ &\le\frac{1}{k_0+k}\cdot\frac{1}{k_0-1}.\end{aligned} \tag{39}$$

To drive the last inequality, we use $\sum_{k'=k_0}^{k}\frac{1}{(k')^2}\le\frac{1}{k_0-1}$. For T_6, we have

$$\begin{aligned}T_6&=\left[\frac{k_0}{k_0+k}\cdot\frac{1}{k_0+0}+\frac{k_0+1}{k_0+k}\cdot\frac{1}{k_0+1}+\cdots\right.\\ &\quad\left.+\frac{k_0+k-1}{k_0+k}\cdot\frac{1}{k_0+k-1}+\frac{1}{k_0+k}\right]\\ &=\frac{k+1}{k_0+k}.\end{aligned} \tag{40}$$

Substituting (38), (39) and (40) into (37), we have

$$\begin{aligned}&\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2\\ &\le\frac{k_0-1}{k_0+k}\|\bar{\boldsymbol{\lambda}}^0-\boldsymbol{\lambda}^*\|^2+\frac{1}{(k_0+k)(k_0-1)}\cdot\frac{4608\delta^2}{\beta^4\epsilon^3u_f^2H^2}\\ &\quad+\frac{k+1}{k_0+k}\cdot\frac{192\delta^2}{\beta^2}\cdot\left(1+\frac{9}{\epsilon^3}\right)\cdot(4\rho^2H+\chi^2).\end{aligned} \tag{41}$$

Applying the triangle inequality into (41), we obtain

$$\begin{aligned}&\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|\\ &\le\sqrt{\frac{k_0-1}{k_0+k}}\|\bar{\boldsymbol{\lambda}}^0-\boldsymbol{\lambda}^*\|+\sqrt{\frac{1}{(k_0+k)(k_0-1)}\cdot\frac{4608\delta^2}{\beta^4\epsilon^3u_f^2H^2}}\\ &\quad+\sqrt{\frac{k+1}{k_0+k}\cdot\frac{192\delta^2}{\beta^2}\cdot\left(1+\frac{9}{\epsilon^3}\right)\cdot(4\rho^2H+\chi^2)}.\end{aligned} \tag{42}$$

By Lemma 4 and using the step size $\gamma^k=\frac{4}{\beta\cdot\frac{H}{J}(k_0+k)}$, we obtain

$$\begin{aligned}\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\bar{\boldsymbol{\lambda}}^{k+1}\|^2&=\|\Lambda^{k+1}-\frac{1}{|\mathcal{H}|}11^\top\Lambda^{k+1}\|_F^2\\ &\le\frac{18(\gamma^{k+1})^2\delta^2H^3}{\epsilon^3\cdot J^2}\\ &=\frac{288\delta^2H}{(k_0+k+1)^2\epsilon^3\beta^2}.\end{aligned} \tag{43}$$

By $(a_1+\cdots+a_H)^2\le H(a_1^2+\cdots+a_H^2)$ and (43), we have

$$\begin{aligned}\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\bar{\boldsymbol{\lambda}}^{k+1}\|&=\sqrt{H}\cdot\sqrt{\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\bar{\boldsymbol{\lambda}}^{k+1}\|^2}\\ &\le\sqrt{H}\cdot\sqrt{\frac{288\delta^2H}{(k_0+k+1)^2\epsilon^3\beta^2}}.\end{aligned} \tag{44}$$

Combining (42) and (44) yields

$$\begin{aligned}&\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\boldsymbol{\lambda}^*\|\\&\le\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\bar{\boldsymbol{\lambda}}^{k+1}\|+H\cdot\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|\\&\le H\cdot\sqrt{\frac{k_0-1}{k_0+k}}\|\bar{\boldsymbol{\lambda}}^0-\boldsymbol{\lambda}^*\|+\sqrt{\frac{1}{(k_0+k)(k_0-1)}\cdot\frac{4608\delta^2}{\beta^4\epsilon^3u_f^2}}\\&\quad+H\cdot\sqrt{\frac{k+1}{k_0+k}\cdot\frac{192\delta^2}{\beta^2}\cdot\left(1+\frac{9}{\epsilon^3}\right)\cdot(4\rho^2H+\chi^2)}\\&\quad+\sqrt{H}\cdot\sqrt{\frac{288\delta^2H}{(k_0+k+1)^2\epsilon^3\beta^2}}.\end{aligned}\tag{45}$$

Taking $k\to+\infty$, we obtain

$$\begin{aligned}&\limsup_{k\to+\infty}\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\boldsymbol{\lambda}^*\|\\&\le\sqrt{\frac{192\delta^2H^2}{\beta^2}\cdot\left(1+\frac{9}{\epsilon^3}\right)\cdot(4\rho^2H+\chi^2)}.\end{aligned}\tag{46}$$

A2 Part b of Theorem 2

Based on $\|\Lambda^{k+1}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+1}\|_F^2\le\frac{18(\gamma^{k+1})^2\delta^2H^3}{\epsilon^3\cdot J^2}$ in Lemma 4 and the fact $\|\Lambda^{k+1}-\frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+1}\|_F^2=\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\bar{\boldsymbol{\lambda}}^{k+1}\|^2$, with a proper decreasing step size $\gamma^k=O(\frac{1}{k})$, taking $k\to+\infty$, we obtain

$$\lim_{k\to+\infty}\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^{k+1}-\bar{\boldsymbol{\lambda}}^{k+1}\|=0.\tag{47}$$

We conclude by summarizing the conditions on the step size γ^k in Theorem 2. It must satisfy $(\gamma^k)^2\cdot\frac{H}{J}-2\gamma^k\alpha\le0$, $\frac{18(\gamma^k)^2}{\epsilon u_f^2J^2}\le\frac{(2-\epsilon)\epsilon^2}{3(3-\epsilon)}$, $1\le\frac{(\gamma^k)^2}{(\gamma^{k+1})^2}\le\frac{2}{1+(1-\epsilon^2)}$, as well as $\gamma^k\le\frac{u_fJ}{2\sqrt{3}}$. The specific step size $\gamma^k=\frac{4}{\beta\cdot\frac{H}{J}(k_0+k)}$ with $k_0\ge\max\{\frac{2}{\alpha\beta},\sqrt{\frac{216(3-\epsilon)}{(2-\epsilon)\epsilon u_f^2H^2\beta^2}},\frac{1}{\sqrt{\frac{2}{1+(1-\epsilon^2)}}-1},\frac{8\sqrt{3}}{u_fH\beta}\}$ satisfies these conditions.

A3 Part c of Theorem 2

The Lagrangian function of (2) is

$$\mathcal{L}(\boldsymbol{\Theta};\boldsymbol{\lambda}):=\frac{1}{H}\sum_{i\in\mathcal{H}}f_i(\boldsymbol{\theta}_i)+\left\langle\boldsymbol{\lambda},\frac{1}{H}\sum_{i\in\mathcal{H}}\boldsymbol{\theta}_i-s\right\rangle.\tag{48}$$

Since $\boldsymbol{\Theta}^*$ is the optimal solution of the primal problem (2), we have $\frac{1}{H}\sum_{i\in\mathcal{H}}\boldsymbol{\theta}_i^*=s$ and $\boldsymbol{\theta}^*\in C$. According to (48), for any dual variable $\boldsymbol{\lambda}$ we have

$$\begin{aligned}\mathcal{L}(\boldsymbol{\Theta}^*;\boldsymbol{\lambda})&=\frac{1}{H}\sum_{i\in\mathcal{H}}f_i(\boldsymbol{\theta}_i^*)+\left\langle\boldsymbol{\lambda},\frac{1}{H}\sum_{i\in\mathcal{H}}\boldsymbol{\theta}_i^*-s\right\rangle\\&=f(\boldsymbol{\Theta}^*).\end{aligned}\tag{49}$$

By Assumption 2, the duality gap is zero. According to (23), for any $\boldsymbol{\lambda}$ we obtain $g(\boldsymbol{\lambda}^*)=-f(\boldsymbol{\Theta}^*)=-\mathcal{L}(\boldsymbol{\Theta}^*;\boldsymbol{\lambda})$. Now we introduce a vector ${}^\dagger\boldsymbol{\Theta}^{k+1}:=[{}^\dagger\boldsymbol{\theta}_1^{k+1};\cdots;{}^\dagger\boldsymbol{\theta}_H^{k+1}]$, where ${}^\dagger\boldsymbol{\theta}_i^{k+1}=\arg\min_{\boldsymbol{\theta}_i\in C_i}\{\boldsymbol{\theta}_i^\top\bar{\boldsymbol{\lambda}}^{k+1}+f_i(\boldsymbol{\theta}_i)\}$ and $\bar{\boldsymbol{\lambda}}^{k+1}:=\frac{1}{H}\sum_{i\in\mathcal{H}}\boldsymbol{\lambda}_i^{k+1}$. Therefore, we have

$$\begin{aligned}&g(\bar{\boldsymbol{\lambda}}^{k+1})-g(\boldsymbol{\lambda}^*)\\&=-\inf_{\boldsymbol{\Theta}\in C}\mathcal{L}\left(\boldsymbol{\Theta};\bar{\boldsymbol{\lambda}}^{k+1}\right)+\mathcal{L}\left(\boldsymbol{\Theta}^*;\bar{\boldsymbol{\lambda}}^{k+1}\right)\\&=-\mathcal{L}\left({}^\dagger\boldsymbol{\Theta}^{k+1};\bar{\boldsymbol{\lambda}}^{k+1}\right)+\mathcal{L}\left(\boldsymbol{\Theta}^*;\bar{\boldsymbol{\lambda}}^{k+1}\right).\end{aligned}\tag{50}$$

Assumption 1 shows that the local cost function $f_i(\cdot)$ is u_f-strongly convex. Further using the definition of $\mathcal{L}(\boldsymbol{\Theta};\boldsymbol{\lambda})$ in (48), we know that $\mathcal{L}(\boldsymbol{\Theta};\boldsymbol{\lambda})$ is u_f-strongly convex with respect to $\boldsymbol{\Theta}$. Therefore, we have

$$\begin{aligned}&\mathcal{L}\left(\boldsymbol{\Theta}^*;\bar{\boldsymbol{\lambda}}^{k+1}\right)-\mathcal{L}\left({}^\dagger\boldsymbol{\Theta}^{k+1};\bar{\boldsymbol{\lambda}}^{k+1}\right)\\&\ge\nabla\mathcal{L}^\top\left({}^\dagger\boldsymbol{\Theta}^{k+1};\bar{\boldsymbol{\lambda}}^{k+1}\right)(\boldsymbol{\Theta}^*-{}^\dagger\boldsymbol{\Theta}^{k+1})+\frac{u_f}{2}\|{}^\dagger\boldsymbol{\Theta}^{k+1}-\boldsymbol{\Theta}^*\|^2.\end{aligned}\tag{51}$$

Combining (50) and (51), we obtain

$$\begin{aligned}&g(\bar{\boldsymbol{\lambda}}^{k+1})-g(\boldsymbol{\lambda}^*)\\&\ge\nabla\mathcal{L}^\top\left({}^\dagger\boldsymbol{\Theta}^{k+1};\bar{\boldsymbol{\lambda}}^{k+1}\right)(\boldsymbol{\Theta}^*-{}^\dagger\boldsymbol{\Theta}^{k+1})+\frac{u_f}{2}\|{}^\dagger\boldsymbol{\Theta}^{k+1}-\boldsymbol{\Theta}^*\|^2\\&\ge\frac{u_f}{2}\|{}^\dagger\boldsymbol{\Theta}^{k+1}-\boldsymbol{\Theta}^*\|^2.\end{aligned}\tag{52}$$

To drive the last inequality, we use optimality condition of ${}^\dagger\boldsymbol{\theta}_i^{k+1}=\arg\min_{\boldsymbol{\theta}_i\in C_i}\{\boldsymbol{\theta}_i^\top\bar{\boldsymbol{\lambda}}^{k+1}+f_i(\boldsymbol{\theta}_i)\}$ [38, Proposition 2.1.2].

According to Lemma 2, $g_i(\boldsymbol{\lambda})$ is smooth with constant $\frac{1}{Hu_f}$. Therefore, function $g(\boldsymbol{\lambda})=\sum_{i\in\mathcal{H}}g_i(\boldsymbol{\lambda})$ is smooth with constant $\frac{1}{u_f}$. This fact leads to

$$\begin{aligned}&g(\bar{\boldsymbol{\lambda}}^{k+1})-g(\boldsymbol{\lambda}^*)\\&\le\nabla g^\top(\boldsymbol{\lambda}^*)(\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*)+\frac{1}{2u_f}\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2\\&=\frac{1}{2u_f}\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2.\end{aligned}\tag{53}$$

To drive the last equality, we use the fact that $\nabla g(\boldsymbol{\lambda}^*)=0$. Combining (52) and (53), we have

$$\|{}^\dagger\boldsymbol{\Theta}^{k+1}-\boldsymbol{\Theta}^*\|^2\le\frac{1}{(u_f)^2}\|\bar{\boldsymbol{\lambda}}^{k+1}-\boldsymbol{\lambda}^*\|^2.\tag{54}$$

Combining (42) and (54), we obtain

$$\begin{aligned}&\|{}^\dagger\boldsymbol{\Theta}^{k+1}-\boldsymbol{\Theta}^*\|\le\frac{1}{u_f}\\&\cdot\left[\sqrt{\frac{k_0-1}{k_0+k}}\|\bar{\boldsymbol{\lambda}}^0-\boldsymbol{\lambda}^*\|+\sqrt{\frac{1}{(k_0+k)(k_0-1)}\cdot\frac{288\delta^2}{\beta^4\epsilon^3u_f^2H^2}}\right.\\&\left.+\sqrt{\frac{k+1}{k_0+k}\cdot\frac{192\delta^2}{\beta}\cdot\left(1+\frac{3}{\epsilon^3}\right)\cdot(4\rho^2H+\chi^2)}\right].\end{aligned}\tag{55}$$

Taking $k\to+\infty$, we obtain

$$\limsup_{k\to+\infty}\|{}^\dagger\boldsymbol{\Theta}^{k+1}-\boldsymbol{\Theta}^*\|$$

$$\leq \frac{1}{u_f} \cdot \sqrt{\frac{192\delta^2}{\beta} \cdot \left(1 + \frac{3}{\epsilon^3}\right) \cdot (4\rho^2 H + \chi^2)}. \quad (56)$$

According to Assumption 1, $f_i(\theta_i)$ is u_f-strongly convex. By the conjugate correspondence theorem in [37], the conjugate function $F_i^*(\lambda) = \max_{\boldsymbol{\theta}_i \in C_i}\{\lambda^\top \theta_i - f_i(\theta_i)\}$ is $\frac{1}{u_f}$-smooth. In consequence, the gradient $\nabla F_i^*(-\lambda) = \arg\min_{\boldsymbol{\theta}_i \in C_i}\{\lambda^\top \theta_i + f_i(\theta_i)\}$ is $\frac{1}{u_f}$-Lipschitz continuous. According to the definition of Lipschitz continuity, we have

$$\|\nabla F_i^*(\lambda_i^{k+1}) - \nabla F_i^*(\bar{\lambda}^{k+1})\| \leq \frac{1}{u_f}\|\lambda_i^{k+1} - \bar{\lambda}^{k+1}\|. \quad (57)$$

Based on ${}^\dagger\theta_i^{k+1} = \arg\min_{\boldsymbol{\theta}_i \in C_i}\{\theta_i^\top \bar{\lambda}^{k+1} + f_i(\theta_i)\}$ and $\theta_i^{k+1} = \arg\min_{\boldsymbol{\theta}_i \in C_i}\{\theta_i^\top \lambda_i^{k+1} + f_i(\theta_i)\}$, we obtain $\theta_i^{k+1} = \nabla F_i^*(\lambda_i^{k+1})$ and ${}^\dagger\theta_i^{k+1} = \nabla F_i^*(\bar{\lambda}^{k+1})$. Substituting them into (57), we have

$$\|\theta_i^{k+1} - {}^\dagger\theta_i^{k+1}\| \leq \frac{1}{u_f}\|\lambda_i^{k+1} - \bar{\lambda}^{k+1}\|. \quad (58)$$

Combining (58), ${}^\dagger\Theta^{k+1} := [{}^\dagger\theta_1^{k+1}; \cdots; {}^\dagger\theta_H^{k+1}]$ and $\Theta^{k+1} := [\theta_1^{k+1}; \cdots; \theta_H^{k+1}]$, we obtain

$$\|\Theta^{k+1} - {}^\dagger\Theta^{k+1}\| \leq \frac{1}{u_f}\sum_{i \in \mathcal{H}}\|\lambda_i^{k+1} - \bar{\lambda}^{k+1}\|. \quad (59)$$

Combining (44) and (59), we have

$$\|\Theta^{k+1} - {}^\dagger\Theta^{k+1}\| \leq \frac{\sqrt{H}}{u_f} \cdot \sqrt{\frac{288\delta^2 H}{(k_0 + k + 1)^2 \epsilon^3 \beta^2}}. \quad (60)$$

Taking $k \to +\infty$, we obtain

$$\lim_{k \to +\infty}\|\Theta^{k+1} - {}^\dagger\Theta^{k+1}\| = 0. \quad (61)$$

Combining (56) and (61) yields

$$\begin{aligned} &\limsup_{k \to +\infty}\|\Theta^{k+1} - \Theta^*\| \\ &\leq \limsup_{k \to +\infty}\|\Theta^{k+1} - {}^\dagger\Theta^{k+1}\| + \limsup_{k \to +\infty}\|{}^\dagger\Theta^{k+1} - \Theta^*\| \\ &\leq \frac{1}{u_f} \cdot \sqrt{\frac{192\delta^2}{\beta} \cdot \left(1 + \frac{3}{\epsilon^3}\right) \cdot (4\rho^2 H + \chi^2)}. \end{aligned} \quad (62)$$

A4 Supporting Lemmas

Lemma 1: Under Assumption 1, for any $\lambda \in \mathbb{R}^D$, the maximum distance between the honest agents' local dual gradients and their average, denoted by $\max_{i \in \mathcal{H}}\|\nabla g_i(\lambda) - \frac{1}{H}\sum_{i \in \mathcal{H}}\nabla g_i(\lambda)\|^2$, is bounded by some positive constant δ^2.

Proof: See Supplementary A.

Lemma 2: Under Assumption 1, for any honest agent $i \in \mathcal{H}$, the local dual function $g_i(\lambda)$ is strongly convex with constant $\frac{1}{HL_f}$ and smooth with constant $\frac{1}{Hu_f}$.

Proof: See Supplementary B.

Lemma 3: Define a matrix $\Lambda^{k+\frac{1}{2}} = [\cdots, \lambda_i^{k+\frac{1}{2}}, \cdots] \in \mathbb{R}^{H \times D}$ that collects the dual variables $\lambda_i^{k+\frac{1}{2}}$ of all honest agents $i \in \mathcal{H}$ generated by Algorithm 2. Under Assumption 1, we have

$$\begin{aligned} &\|\Lambda^{k+\frac{1}{2}} - \frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\|_F^2 \\ &\leq \left(\frac{1}{1-v} + \frac{6(\gamma^k)^2}{v \cdot u_f^2 J^2}\right)\|\Lambda^k - \frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^k\|_F^2 + \frac{3(\gamma^k)^2\delta^2 H^3}{v \cdot J^2}, \end{aligned} \quad (63)$$

where v is any positive constant in (0,1). If $v = \frac{1}{2}$ and the step size $\gamma^k \leq \frac{u_f J}{2\sqrt{3}}$, this further yields

$$\begin{aligned} &\|\Lambda^{k+\frac{1}{2}} - \frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+\frac{1}{2}}\|_F^2 \\ &\leq 3\|\Lambda^k - \frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^k\|_F^2 + \frac{6(\gamma^k)^2\delta^2 H^3}{J^2}. \end{aligned} \quad (64)$$

Proof: See Supplementary C.

Lemma 4: Define a matrix $\Lambda^{k+1} = [\cdots, \lambda_i^{k+1}, \cdots] \in \mathbb{R}^{H \times D}$ that collects the dual variables λ_i^{k+1} of all honest agents $i \in \mathcal{H}$ generated by Algorithm 2. Suppose that the robust aggregation rules AGG_i satisfy (14) in Definition 1. Under Assumptions 1 and 3, if the contraction constant ρ satisfies $\rho < \frac{1-\kappa}{8\sqrt{H}}$, we have

$$\|\Lambda^{k+1} - \frac{1}{H}\mathbf{1}\mathbf{1}^\top\Lambda^{k+1}\|_F^2 \leq \frac{18(\gamma^{k+1})^2\delta^2 H^3}{\epsilon^3 \cdot J^2}, \quad (65)$$

where $\epsilon := 1 - \kappa - 8\rho\sqrt{H}$.

Proof: See Supplementary D.

Lemma 5: Suppose that for any integer $k \geq 0$, a sequence $\{\gamma^k\}$ satisfies

$$1 \leq \frac{(\gamma^k)^2}{(\gamma^{k+1})^2} \leq \frac{2}{1 + \psi_1} \quad (66)$$

for some $\psi_1 \in (0,1)$, and another sequence $\{y^k\}$ satisfies

$$y^{k+1} \leq \psi_1 y^k + \psi_2(\gamma^k)^2 \quad \text{and} \quad y^0 \leq \psi_2(\gamma^0)^2 \quad (67)$$

for some $\psi_1 \in (0,1)$ and $\psi_2 \geq 0$. Then, y^k is upper-bounded by

$$y^k \leq \frac{2\psi_2}{1 - \psi_1}(\gamma^k)^2. \quad (68)$$

Proof: See Supplementary E.

REFERENCES

[1] R. Wang, Q. Ling, and Z. Tian, "D3: Dual-domain defenses for Byzantine-resilient decentralized resource allocation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2024, pp. 9331–9335.

[2] J. Koshal, A. Nedic, and U. V. Shanbhag, "Multiuser optimization: Distributed algorithms and error analysis," *SIAM J. Optim.*, vol. 21, no. 3, pp. 1046–1081, 2011.

[3] B. Turan, C. A. Uribe, H. Wai, and M. Alizadeh, "Resilient primal-dual optimization algorithms for distributed resource allocation," *IEEE Trans. Control Netw. Syst.*, vol. 8, no. 1, pp. 282–294, Mar. 2021.

[4] Z. Fan et al., "Smart grid communications: Overview of research challenges, solutions, and standardization activities," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 21–38, First Quarter 2013.

[5] M. Noor-A-Rahim, Z. Liu, H. Lee, G. G. M. N. Ali, D. Pesch, and P. Xiao, "A survey on resource allocation in vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 701–721, Feb. 2022.

[6] A. Ahmad, S. Ahmad, M. H. Rehmani, and N. U. Hassan, "A survey on radio resource allocation in cognitive radio sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 888–917, Second Quarter 2015.

[7] S. Liang, X. Zeng, G. Chen, and Y. Hong, "Distributed sub-optimal resource allocation via a projected form of singular perturbation," *Automatica*, vol. 121, 2020, Art. no. 109180.
[8] W. Jia, N. Liu, and S. Qin, "An adaptive continuous-time algorithm for nonsmooth convex resource allocation optimization," *IEEE Trans. Autom. Control*, vol. 67, no. 11, pp. 6038–6044, Nov. 2022.
[9] 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.*, vol. 51, no. 2, pp. 1097–1106, Feb. 2021.
[10] K. Lu, H. Xu, and Y. Zheng, "Distributed resource allocation via multi-agent systems under time-varying networks," *Automatica*, vol. 136, 2022, Art. no. 110059.
[11] H. Lakshmanan and D. P. D. Farias, "Decentralized resource allocation in dynamic networks of agents," *SIAM J. Optim.*, vol. 19, no. 2, pp. 911–940, 2008.
[12] 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.
[13] E. Ghadimi, I. Shames, and M. Johansson, "Multi-step gradient methods for networked optimization," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5417–5429, Nov. 2013.
[14] T. T. Doan and C. L. Beck, "Distributed resource allocation over dynamic networks with uncertainty," *IEEE Trans. Autom. Control*, vol. 66, no. 9, pp. 4378–4384, Sep. 2021.
[15] 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.
[16] J. Zhang, K. You, and K. Cai, "Distributed dual gradient tracking for resource allocation in unbalanced networks," *IEEE Trans. Signal Process.*, vol. 68, pp. 2186–2198, 2020.
[17] A. Nedić, A. Olshevsky, and W. Shi, "Improved convergence rates for distributed resource allocation," in *Proc. Conf. Decis. Control*, 2018, pp. 172–177.
[18] S. A. Alghunaim, K. Yuan, and A. H. Sayed, "A proximal diffusion strategy for multiagent optimization with sparse affine constraints," *IEEE Trans. Autom. Control*, vol. 65, no. 11, pp. 4554–4567, Nov. 2020.
[19] L. Lamport, R. E. Shostak, and M. C. Pease, "The Byzantine generals problem," *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, pp. 382–401, 1982.
[20] Z. Yang, A. Gang, and W. U. Bajwa, "Adversary-resilient distributed and decentralized statistical inference and machine learning: An overview of recent advances under the Byzantine threat model," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 146–159, May 2020.
[21] D. Yin, Y. Chen, K. Ramchandran, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 5650–5659.
[22] Y. Chen, L. Su, and J. Xu, "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent," *ACM Meas. Anal. Comput. Syst.*, vol. 1, no. 2, pp. 1–25, 2017.
[23] C. Xie, O. Koyejo, and I. Gupta, "Generalized Byzantine-tolerant SGD," 2018, *arXiv: 1802.10116*.
[24] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 118–128.
[25] Z. Wu, T. Chen, and Q. Ling, "Byzantine-resilient decentralized stochastic optimization with robust aggregation rules," *IEEE Trans. Signal Process.*, vol. 71, pp. 3179–3195, 2023.
[26] L. Su and N. H. Vaidya, "Byzantine-resilient multiagent optimization," *IEEE Trans. Autom. Control*, vol. 66, no. 5, pp. 2227–2233, May 2021.
[27] Z. Yang and W. U. Bajwa, "ByRDiE: Byzantine-resilient distributed coordinate descent for decentralized learning," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 4, pp. 611–627, Dec. 2019.
[28] S. Sundaram and B. Gharesifard, "Distributed optimization under adversarial nodes," *IEEE Trans. Autom. Control*, vol. 64, no. 3, pp. 1063–1076, Mar. 2019.
[29] L. Su and S. Shahrampour, "Finite-time guarantees for Byzantine-resilient distributed state estimation with noisy measurements," *IEEE Trans. Autom. Control*, vol. 65, no. 9, pp. 3758–3771, Sep. 2020.
[30] C. Fang, Z. Yang, and W. U. Bajwa, "BRIDGE: Byzantine-resilient decentralized gradient descent," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 8, pp. 610–626, 2022.
[31] W. Abbas, M. Shabbir, J. Li, and X. Koutsoukos, "Resilient distributed vector consensus using centerpoint," *Automatica*, vol. 136, 2022, Art. no. 110046.
[32] K. Kuwarancharoen, L. Xin, and S. Sundaram, "Byzantine-resilient distributed optimization of multi-dimensional functions," in *Proc. Amer. Control Conf.*, 2020, pp. 4399–4404.
[33] L. He, S. P. Karimireddy, and M. Jaggi, "Byzantine-robust decentralized learning via self-centered clipping," 2022, *arXiv: 2202.01545*.
[34] R. Wang, Y. Liu, and Q. Ling, "Byzantine-resilient resource allocation over decentralized networks," *IEEE Trans. Signal Process.*, vol. 70, pp. 4711–4726, 2022.
[35] Q. Li, D. Gao, H. Zhang, Z. Wu, and F. Wang, "Consensus-based distributed economic dispatch control method in power systems," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 941–954, Jan. 2019.
[36] H. Li, Z. Wang, G. Chen, and Z. Dong, "Distributed robust algorithm for economic dispatch in smart grids over general unbalanced directed networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 7, pp. 4322–4332, Jul. 2020.
[37] A. Beck, *First-order Methods in Optimization*. Philadelphia, PA, USA: SIAM, 2017.
[38] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Sci., 2016.
[39] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
[40] B. Johansson, T. Keviczky, M. Johansson, and K. H. Johansson, "Subgradient methods and consensus algorithms for solving convex optimizationn problems," in *Proc. 47th IEEE Conf. Decis. Control*, 2008, pp. 4185–4190.
[41] H. Ye, H. Zhu, and Q. Ling, "On the tradeoff between privacy preservation and Byzantine-robustness in decentralized learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2024, pp. 9336–9340.
[42] J. Peng, W. Li, and Q. Ling, " Byzantine-robust decentralized stochastic optimization over static and time-varying networks," *Signal Process.*, vol. 183, 2021, Art. no. 108020.
[43] B. Kailkhura, S. Brahma, and P. K. Varshney, "Data falsification attacks on consensus-based detection systems," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 1, pp. 145–158, Mar. 2017.
[44] F. P. Kelly, A. K. Maulloo, and D. K. Tan, "Rate control for communication networks: Shadow prices, proportional fairness, and stability," *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, 1998.
[45] R. Wang, Q. Ling, and Z. Tian, "D3: Dual-domain defenses for Byzantine-resilient decentralized resource allocation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2024, pp. 9331–9335.
[46] "IEEE 118 bus system," 2015. [Online]. Available: https://www.al-roomi.org/power-flow/118-bus-system
[47] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXTRA: An exact first-order algorithm for decentralized consensus optimization," *SIAM J. Optim.*, vol. 25, no. 2, pp. 944–966, 2015.
[48] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM J. Optim.*, vol. 26, no. 3, pp. 1835–1854, 2016.

Runhua Wang (Member, IEEE) received the B.E. degree in network engineering from Huaibei Normal University, Huaibei, China, and the M.E. degree in software engineering from Central South University, Changsha, China. She is currently working toward the Ph.D. degree with Sun Yat-sen University, Guangzhou, China. Her research interests include resource allocation and distributed optimization.

Qing Ling (Senior Member, IEEE) received the B.E. degree in automation and the Ph.D. degree in control theory and control engineering from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively. He was a Postdoctoral Research Fellow with the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI, USA, from 2006 to 2009, and Associate Professor with the Department of Automation, University of Science and Technology of China, from 2009 to 2017. He is currently a Professor with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China. His research interests include distributed and decentralized optimization and its application in machine learning. He was the recipient of the 2017 IEEE Signal Processing Society Young Author Best Paper Award as a supervisor. He was a TPC Chair of the 2023 IEEE SPAWC Workshop. He was an Associate Editor and Senior Area Editor of IEEE SIGNAL PROCESSING LETTERS, and Associate Editor for IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT. He is an Associate Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING.

Zhi Tian (Fellow, IEEE) is currently a Professor with the Electrical and Computer Engineering Department, George Mason University, Fairfax, VA, USA, since 2015. She was with the Faculty of Michigan Technological University, Houghton, MI, USA, from 2000 to 2014. She was a Program Director with the US National Science Foundation, Alexandria, VA, USA, from 2012 to 2014. Her research interests include the areas of statistical signal processing, wireless communications, and distributed machine learning. Her current research focuses on distributed network optimization and learning, wireless spectrum sensing and millimeter-wave systems. She was an IEEE Distinguished Lecturer for the IEEE Communications Society and IEEE Vehicular Technology Society. She was an Associate Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and IEEE TRANSACTIONS ON SIGNAL PROCESSING. She was General Co-Chair of the 2016 IEEE GlobalSIP Conference and 2023 IEEE SPAWC Workshop. She was a Member-of-Large of the Board of Governors of the IEEE Signal Processing Society for the term of 2019–2021. She was the recipient of the IEEE Communications Society TCCN Publication Award in 2018.