ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10446020

D3: DUAL-DOMAIN DEFENSES FOR BYZANTINE-RESILIENT DECENTRALIZED RESOURCE ALLOCATION

Runhua Wang[1] *Qing Ling*[1] *Zhi Tian*[2]

[1]Sun Yat-Sen University [2]George Mason University

ABSTRACT

This paper considers 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 converge to a neighborhood 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

1. INTRODUCTION

Decentralized resource allocation has found wide applications in various fields, such as smart grids, transportation systems, to name a few [1,2]. Mathematically speaking, it minimizes the average cost of decentralized agents subject to local and global resource constraints, where the optimization variable is the resource allocation strategy.

Decentralized Resource Allocation Algorithms. In a decentralized resource allocation problem, the primary challenge is to satisfy the global resource constraint. Weighted gradient methods have been proposed to guarantee global constraint satisfaction with the aid of feasible initialization [3–5], but they are sensitive to perturbations. Among them, [3] considers time-varying networks, while [4] considers static networks. The work of [5] utilizes historical information to accelerate the algorithm. On the other hand, primal-dual algorithms handle the global resource constraint via introducing a dual variable [6–10]. The works of [6, 7] develop decentralized Lagrangian methods, which precisely solve the primal sub-problems while perform a dual gradient step at each iteration. The work of [8] employs a 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 [9, 10].

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 attack model and term the malicious agents as Byzantine agents [11, 12]. We briefly review some general Byzantine-resilient decentralized optimization algorithms and few Byzantine-resilient resource allocation algorithms, as follows.

Byzantine-resilient Algorithms. In a general Byzantine-resilient decentralized optimization problem, honest agents cooperate for reaching 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). A common feature in the existing algorithms is to let each honest agent aggregate possibly malicious messages (namely, optimization variables) received from its neighbors in a robust manner. When the cost functions are deterministic and the optimization variable is a scalar, [13, 14] uses 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. For high-dimensional problems, [15, 16] extends TM to coordinate-wise TM (CTM), such that each honest agent performs the TM operation at each dimension. When the cost functions are stochastic, TM and CTM are also applicable. Besides, the work of [17] 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 [18] 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 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. To fill this gap, [19] proposes a primal-dual Byzantine-resilient resource allocation algorithm, 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 [20]. 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 let each honest agent fuse the received neighboring dual variables with a properly designed robust aggregation rule, including but not limited to CTM, IOS and SCC.

C2) Compared with BREDA that defends against Byzantine attacks

in the primal domain [20], 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 pairs, 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.

2. PROBLEM FORMULATION

We consider a decentralized resource allocation problem that involves a network of autonomous agents. The network is modeled as an undirected, connected graph $\tilde{\mathcal{G}}(\mathcal{J},\tilde{\mathcal{E}})$ with the set of vertices $\mathcal{J}:=\{1,\cdots,J\}$ and the set of edges $\tilde{\mathcal{E}}$. If $(i,j)\in\tilde{\mathcal{E}}$, then the two agents i and j are neighbors and can communicate with each other. For agent i, define the set of neighbors as $\mathcal{N}_i=\{j\mid(i,j)\in\tilde{\mathcal{E}}\}$. Each agent i possesses a local cost function $f_i(\boldsymbol{\theta}_i)$, where $\boldsymbol{\theta}_i\in\mathbb{R}^D$ stands for the amount of local resources and belongs to a compact, convex set C_i. The average amount of local resources, denoted as $\frac{1}{J}\sum_{i=1}^{J}\boldsymbol{\theta}_i$, equals to a constant vector $\boldsymbol{s}\in\mathbb{R}^D$. When all the agents are honest, the decentralized resource allocation problem is formulated as

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

where $\tilde{\boldsymbol{\Theta}}=[\boldsymbol{\theta}_1,\cdots,\boldsymbol{\theta}_J]\in\mathbb{R}^{JD}$ concatenates all the local variables and $\tilde{C}$ is the Cartesian product of C_i for all $i\in\mathcal{J}$.

When some of the agents are Byzantine, 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 $\tilde{\mathcal{G}}(\mathcal{J},\tilde{\mathcal{E}})$, where $\mathcal{E}=\{(i,j)\in\tilde{\mathcal{E}};i,j\in\mathcal{H}\}$ 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

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

where $\boldsymbol{\Theta}=[\boldsymbol{\theta}_1,\cdots,\boldsymbol{\theta}_H]\in\mathbb{R}^{HD}$ concatenates all the local variables of the honest agents and C is the Cartesian product of C_i for all $i\in\mathcal{H}$.

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, we focus on developing Byzantine-resilient decentralized resource allocation algorithms to approximately solve (2).

3. ATTACK-FREE DECENTRALIZED RESOURCE ALLOCATION

Algorithm Review. We begin with reviewing an attack-free decentralized resource allocation algorithm, which operates in the dual domain, to solve (1). The dual problem of (1) is given by

$$\min_{\boldsymbol{\lambda}\in\mathbb{R}^D}\sum_{i\in\mathcal{J}}\tilde{g}_i(\boldsymbol{\lambda}),\tag{3}$$

where $\boldsymbol{\lambda}$ is the dual variable, $\tilde{g}_i(\boldsymbol{\lambda}):=\frac{1}{J}F_i^*(-\boldsymbol{\lambda})+\frac{1}{J}\boldsymbol{\lambda}^\top\boldsymbol{s}$ and $F_i^*(-\boldsymbol{\lambda}):=\max_{\boldsymbol{\theta}_i\in C_i}\{-\boldsymbol{\lambda}^\top\boldsymbol{\theta}_i-f_i(\boldsymbol{\theta}_i)\}$.

Because (3) is separable across the agents $i\in\mathcal{J}$, it can be solved through a decentralized gradient method [6, 21, 22]. Introducing local dual variable $\boldsymbol{\lambda}_i$ to each agent i, we have the following updates

$$\boldsymbol{\theta}_i^k=\arg\min_{\boldsymbol{\theta}_i\in C_i}\{\boldsymbol{\theta}_i^\top\boldsymbol{\lambda}_i^k+f_i(\boldsymbol{\theta}_i)\},\tag{4}$$

$$\boldsymbol{\lambda}_i^{k+\frac{1}{2}}=\boldsymbol{\lambda}_i^k-\gamma^k\nabla\tilde{g}_i(\boldsymbol{\lambda}_i^k)=\boldsymbol{\lambda}_i^k-\gamma^k(\frac{1}{J}\boldsymbol{s}-\frac{1}{J}\boldsymbol{\theta}_i^k),\tag{5}$$

$$\boldsymbol{\lambda}_i^{k+1}=\sum_{j\in\mathcal{N}_i\cup\{i\}}\tilde{e}_{ij}\boldsymbol{\lambda}_j^{k+\frac{1}{2}}.\tag{6}$$

Therein, $\gamma^k>0$ is the step size and $\tilde{e}_{ij}\geq0$ is the weight assigned by agent i to agent j. Note that $\tilde{e}_{ij}>0$ if and only if $(i,j)\in\tilde{\mathcal{E}}$ or $i=j$. We collect these weights in $\tilde{E}=[\tilde{e}_{ij}]\in\mathbb{R}^{J\times J}$, which is assumed to be doubly stochastic.

Failure of (4)–(6) under Byzantine Attacks. When all the agents are honest, the decentralized resource allocation algorithm outlined in (4)–(6) can effectively solve (1) [6,21,22]. 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

$$\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}=\begin{cases}\boldsymbol{\lambda}_j^{k+\frac{1}{2}}, & j\in\mathcal{H},\\ *, & j\in\mathcal{B}.\end{cases}\tag{7}$$

By sending the malicious messages, the Byzantine agents can easily prevent the honest agents from obtaining the optimal dual variable and corresponding resource allocation strategy.

4. BYZANTINE-RESILIENT DECENTRALIZED RESOURCE ALLOCATION

Algorithm Development. As we have shown in Section 3, the decentralized resource allocation algorithm outlined in (4)–(6) fails in the presence of Byzantine attacks. This is due to the vulnerability of the weighted average aggregation in (6) to Byzantine attacks. To address this issue, we replace the weighted average aggregation with some 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

$$\boldsymbol{\theta}_i^k=\arg\min_{\boldsymbol{\theta}_i\in C_i}\{\boldsymbol{\theta}_i^\top\boldsymbol{\lambda}_i^k+f_i(\boldsymbol{\theta}_i)\},\tag{8}$$

$$\boldsymbol{\lambda}_i^{k+\frac{1}{2}}=\boldsymbol{\lambda}_i^k-\gamma^k(\frac{1}{J}\boldsymbol{s}-\frac{1}{J}\boldsymbol{\theta}_i^k),\tag{9}$$

$$\boldsymbol{\lambda}_i^{k+1}=AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}},\{\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i}),\tag{10}$$

where $AGG_i(\cdot)$ denotes a robust aggregation rule of honest agent i.

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 (8)–(10) to converge to a neighborhood 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 $AGG_i(\boldsymbol{\lambda}_i^{k+\frac{1}{2}}, \{\check{\boldsymbol{\lambda}}_j^{k+\frac{1}{2}}\}_{j\in\mathcal{N}_i})$ is close to a proper weighted average of the messages from its honest neighbors and its own local dual variable. We denote such a weighted average 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 [17, 23] to characterize a set of robust aggregation rules with a weight matrix and a contraction constant.

Definition 1. *Consider a set of robust aggregation rules denoted as $\{AGG_i\}_{i\in\mathcal{H}}$. 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*

$$\|AGG_i(\boldsymbol{\lambda}_i, \{\check{\boldsymbol{\lambda}}_j\}_{j\in\mathcal{N}_i}) - \bar{\boldsymbol{\lambda}}_i\| \leq \rho \max_{j\in(\mathcal{N}_i\cap\mathcal{H})\cup i} \|\boldsymbol{\lambda}_j - \bar{\boldsymbol{\lambda}}_i\| \quad (11)$$

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

It has been shown in [23] that CTM, IOS, SCC, as well as a number of other robust aggregation rules, all satisfy Definition 1.

Advantages over BREDA. Our proposed algorithms have several advantages over BREDA [20]: 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 [24], 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.

5. CONVERGENCE ANALYSIS

Here we establish convergence of the Byzantine-free and Byzantine-resilient decentralized resource allocation algorithms, outlined in (4)–(6) and (8)–(10), respectively. Due to the page limit, we omit the detailed proofs. 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 exists $\tilde{\boldsymbol{\Theta}}$ and $\boldsymbol{\Theta}$ in the relative interiors of $\tilde{C}$ and C, such that the constraints $\frac{1}{J}\sum_{i\in\mathcal{J}}\boldsymbol{\theta}_i = \boldsymbol{s}$ and $\frac{1}{H}\sum_{i\in\mathcal{H}}\boldsymbol{\theta}_i = \boldsymbol{s}$ satisfy, respectively.*

Assumption 3. *The graphs $\tilde{\mathcal{G}}(\mathcal{J}, \tilde{\mathcal{E}})$ and $\mathcal{G}(\mathcal{J}, \mathcal{E})$ are both undirected and connected. The weight matrices $\tilde{E}$ and E are doubly stochastic and row stochastic, respectively, and satisfy $\tilde{\kappa} := \|\tilde{E} - \frac{1}{J}\tilde{\mathbf{1}}\tilde{\mathbf{1}}^\top\|^2 < 1$ and $\kappa := \|E - \frac{1}{H}\mathbf{1}\mathbf{1}^\top E\|^2 < 1$, where $\tilde{\mathbf{1}} := [1, \cdots, 1] \in \mathbb{R}^J$ and $\mathbf{1} := [1, \cdots, 1] \in \mathbb{R}^H$.*

Attack-free Decentralized Resource Allocation. Denote $(\tilde{\boldsymbol{\Theta}}^*, \tilde{\boldsymbol{\lambda}}^*)$ as the optimal primal-dual pair of (1), in which $\tilde{\boldsymbol{\Theta}}^* \in \mathbb{R}^{JD}$ and $\tilde{\boldsymbol{\lambda}}^* \in \mathbb{R}^D$. The following theorem shows the convergence of the attack-free decentralized allocation algorithm (4)–(6).

Theorem 1. *Consider $\tilde{\boldsymbol{\Theta}}^{k+1}$ and $\{\boldsymbol{\lambda}_i^{k+1}\}_{i\in\mathcal{J}}$ generated by the attack-free decentralized resource allocation algorithm* (4)–(6) *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) $\lim_{k\to+\infty} \|\tilde{\boldsymbol{\Theta}}^{k+1} - \tilde{\boldsymbol{\Theta}}^*\| = 0$,
b) $\lim_{k\to+\infty} \sum_{i\in\mathcal{J}} \|\boldsymbol{\lambda}_i^{k+1} - \tilde{\boldsymbol{\lambda}}^*\| = 0$.

Theorem 1 shows that the local primal and dual variables generated by (4)–(6) converge to their optima. This matches the classical conclusion for the decentralized gradient method [6, 21, 22]. They 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 Byzantine-resilient algorithms.

Byzantine-resilient Decentralized Resource Allocation. 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 (8)–(10).

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* (8)–(10)*. Suppose that Byzantine agents are present but the used robust aggregation rule satisfies* (11) *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) $\limsup_{k\to+\infty} \|\boldsymbol{\Theta}^{k+1} - \boldsymbol{\Theta}^*\| \leq \frac{\Delta}{u_f}$,
b) $\limsup_{k\to+\infty} \sum_{i\in\mathcal{H}} \|\boldsymbol{\lambda}_i^{k+1} - \boldsymbol{\lambda}^*\| \leq \Delta$,
c) $\lim_{k\to+\infty} \sum_{i\in\mathcal{H}} \|\boldsymbol{\lambda}_i^k - \bar{\boldsymbol{\lambda}}^k\| = 0$,
where $\bar{\boldsymbol{\lambda}}^k := \frac{1}{H}\sum_{i\in\mathcal{H}}\boldsymbol{\lambda}_i^k$ and $\Delta \in \mathbb{R}$ is in the order of $O(\rho^2+\chi^2)$, with $\chi^2 := \frac{1}{H}\|E^\top\mathbf{1} - \mathbf{1}\|^2$ quantifying the non-doubly stochasticity of E.

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 (8)–(10) converge to neighborhoods of their optima. Sizes of the neighborhoods are determined by the associated contraction constant ρ and weight matrix E (more precisely, χ^2). Notably, the local dual variables are guaranteed to reach consensus 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. Therefore, 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 during the analysis. Note that when $\rho = 0$ and E is doubly stochastic, Theorem 2 reduces to Theorem 1.

Table 1. BOUNDS OF ρ^2 AND χ^2

	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

Table 2. DUAL CONSENSUS ERRORS $\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^k-\bar{\boldsymbol{\lambda}}^k\|^2$

	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

Our analysis is related to but significantly different from that in [17]. The work of [17] considers a general Byzantine-resilient decentralized stochastic non-convex *optimization* problem, and analyzes robust aggregation rules that satisfy Definition 1 in the primal domain. 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.

6. NUMERICAL EXPERIMENTS

We consider a synthetic and scalar case with $D=1$. Due to the page limit, more results are left to the extended version of this paper [25]. The code is available online.[1] Consider a randomly generated network consisting of $J=100$ agents, where each agent has 15 neighbors. The weight $\tilde{e}_{ij}$ is set to $\frac{1}{16}$ if and only if $(i,j)\in\tilde{\mathcal{E}}$ or $i=j$. The total amount of resources is 5000 such that $\boldsymbol{s}=50$. 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 [4, 7, 8].

We randomly select $B=6$ Byzantine agents, 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 from a Gaussian distribution with mean -30 and variance 5^2. With small-value Gaussian attacks, a Byzantine agent sets its message from a Gaussian distribution with mean -300 and variance 40^2.

We use the attack-free decentralized resource allocation algorithm (4)–(6) and BREDA as baselines. 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.

Figs. 1 illustrates that the attack-free decentralized resource allocation algorithm (4)–(6) 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, dual optimality, cost optimality, and constraint violation. 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 1 according to [23]. From the smallest to the largest are IOS, CTM and SCC, which validates our theoretical findings.

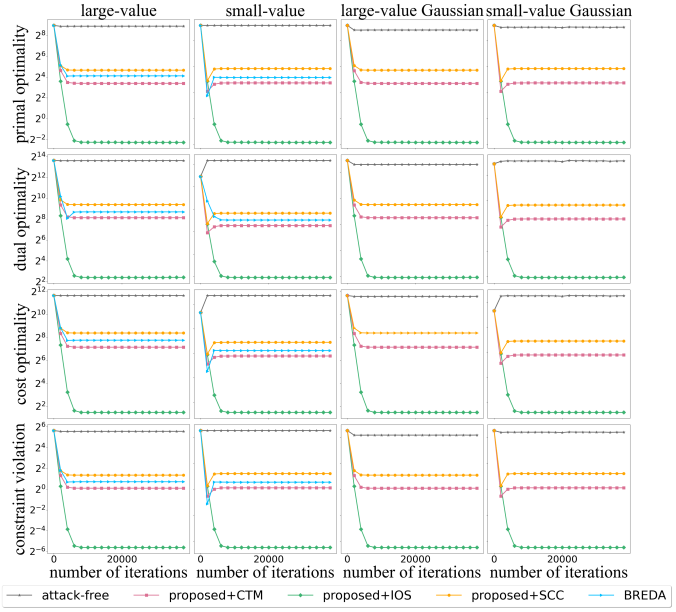

Fig. 1. Primal optimality $\|\boldsymbol{\Theta}^k-\boldsymbol{\Theta}^*\|$, dual optimality $\sum_{i\in\mathcal{H}}\|\boldsymbol{\lambda}_i^k-\boldsymbol{\lambda}^*\|$, cost optimality $\|f(\boldsymbol{\Theta}^k)-f(\boldsymbol{\Theta}^*)\|$ and constraint violation $\|\frac{1}{\mathcal{H}}\sum_{i\in\mathcal{H}}\boldsymbol{\theta}_i^k-\boldsymbol{s}\|$ of the compared algorithms.

From Figs. 1, we find 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 2. No matter the types of Byzantine attacks and robust aggregation rules, the proposed algorithms are all able to achieve nearly perfect dual consensus. By contrast, BREDA cannot guarantee dual consensus. This phenomenon reveals the benefits of the dual-domain defenses.

Conclusions. This paper investigates decentralized resource allocation under Byzantine attacks. We propose a class of Byzantine-resilient algorithms equipped with robust aggregation rules, featured in dual-domain defenses. Given that the robust aggregation rules are properly designed, we prove that the generated primal and dual variables of the honest agents converge to neighborhoods of their optima, while the dual variables are able to reach consensus. The numerical experiments show the resilience of the proposed algorithms to various Byzantine attacks.

Acknowledgement. The work of Qing Ling (corresponding author) is supported in part by NSF China grants 61973324, 12126610 and 62373388, Guangdong Basic and Applied Basic Research Foundation grant 2021B1515020094, as well as R&D project of Pazhou Lab (Huangpu) grant 2023K0606.

[1] https://github.com/RunhuaWang

7. REFERENCES

[1] Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. Kalogridis, M. Sooriyabandara, Z. Zhu, S. Lambotharan, and W. H. Chin, "Smart grid communications: Overview of research challenges, solutions, and standardization activities," IEEE Communications Surveys Tutorials, vol. 15, no. 1, pp. 21–38, 2013.

[2] M. Noor-A-Rahim, Z. Liu, H. Lee, G. G. M. Nawaz Ali, D. Pesch, and P. Xiao, "A survey on resource allocation in vehicular networks," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 2, pp. 701–721, 2022.

[3] H. Lakshmanan, and D. P. D. Farias, "Decentralized resource allocation in dynamic networks of agents," SIAM Journal on Optimization, vol. 19, no. 2, pp. 911–940, 2008.

[4] L. Xiao, and S. Boyd, "Optimal scaling of a gradient method for distributed resource allocation," Journal of Optimization Theory and Applications, vol. 129, no. 3, pp. 469–488, 2006.

[5] E. Ghadimi, I. Shames, and M. Johansson, "Multi-step gradient methods for networked optimization," IEEE Transactions on Signal Processing, vol. 61, no. 21, pp. 5417–5429, 2013.

[6] T. T. Doan, and C. L. Beck, "Distributed resource allocation over dynamic networks with uncertainty," IEEE Transactions on Automatic Control, vol. 66, no. 9, pp. 4378–4384, 2021.

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

[8] J. Zhang, K. You, and K. Cai, "Distributed dual gradient tracking for resource allocation in unbalanced networks," IEEE Transactions on Signal Processing, vol. 68, pp. 2186-2198, 2020.

[9] A. Nedić, A. Olshevsky, and W. Shi, "Improved convergence rates for distributed resource allocation," In Proceedings of CDC, 2018.

[10] S. A. Alghunaim, K. Yuan, and A. H. Sayed, "A proximal diffusion strategy for multiagent optimization with sparse affine constraints," IEEE Transactions on Automatic Control, vol. 65, no. 11, pp. 4554–4567, 2020.

[11] L. Lamport, R. E. Shostak, and M. C. Pease, "The Byzantine generals problem," ACM Transactions on Programming Languages and Systems, vol. 4, no. 3, pp. 382–401, 1982.

[12] 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 Processing Magazine, vol. 37, no. 3, pp. 146–159, 2020.

[13] Z. Yang and W. U. Bajwa, "ByRDiE: Byzantine-resilient distributed coordinate descent for decentralized learning," IEEE Transactions on Signal and Information Processing over Networks, vol. 5, no. 4, pp. 611–627, 2019.

[14] L. Su and N. H. Vaidya, "Byzantine-resilient multiagent optimization," IEEE Transactions on Automatic Control, vol. 66, no. 5, pp. 2227–2233, 2021.

[15] L. Su and S. Shahrampour, "Finite-time guarantees for Byzantine-resilient distributed state estimation with noisy measurements," IEEE Transactions on Automatic Control, vol. 65, no. 9, pp. 3758–3771, 2020.

[16] C. Fang, Z. Yang, and W. U. Bajwa, "BRIDGE: Byzantine-resilient decentralized gradient descent," IEEE Transactions on Signal and Information Processing over Networks, vol. 8, pp. 610–626, 2022.

[17] Z. Wu, T. Chen, and Q. Ling, "Byzantine-resilient decentralized stochastic optimization with robust aggregation rules," IEEE Transactions on Signal Processing, vol. 71, pp. 3179–3195, 2023.

[18] L. He, S. P. Karimireddy, and M. Jaggi, "Byzantine-robust decentralized learning via self-centered clipping," arXiv preprint arXiv: 2202.01545, 2022.

[19] B. Turan, C. A. Uribe, H. Wai, and M. Alizadeh, "Resilient primal-dual optimization algorithms for distributed resource allocation," IEEE Transactions on Control of Network Systems, vol. 8, no. 1, pp. 282–294, 2021.

[20] R. Wang, Y. Liu, and Q. Ling, "Byzantine-resilient resource allocation over decentralized networks," IEEE Transactions on Signal Processing, vol. 70, pp. 4711–4726, 2022.

[21] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," IEEE Transactions on Automatic Control, vol. 54, no. 1, pp. 48–61, 2009.

[22] B. Johansson, T. Keviczky, M. Johansson, and K. H. Johansson, "Subgradient methods and consensus algorithms for solving convex optimizationn problems," In Proceedings of CDC, 2008.

[23] H. Ye, H. Zhu, and Q. Ling, "On the tradeoff between privacy preservation and Byzantine-robustness in decentralized learning," arXiv preprint arXiv: 2308.14606, 2023.

[24] F. P. Kelly, A. K. Maulloo, and D. K. Tan, "Rate control for communication networks: Shadow prices, proportional fairness, and stability," Journal of the Operational Research Society, vol. 49, no. 3, pp. 237–252, 1998.

[25] R. Wang, Q. Ling, and Z. Tian, "Dual-domain defenses for Byzantine-resilient decentralized resource allocation," arXiv preprint arXiv: 2310.05698, 2023.