2017 13th IEEE International Conference on Control & Automation (ICCA)
July 3-6, 2017. Ohrid, Macedonia

Resilient Distributed Optimization Algorithm against Adversary Attacks

Chengcheng Zhao[1], Jianping He[2], and Qing-Guo WANG[3]

Abstract— **In the distributed optimization, multiple agents aim to minimize the average of all local cost functions corresponding to one decision variable. Recently, the resilient algorithms for distributed optimization against attacks have received some attention, where it is assumed that the maximum number of tolerable attacks is strictly limited by the network connectivity. To relax this assumption, in this paper, we propose a resilient distribution optimization algorithm by exploiting the trusted agents, which cannot be compromised by adversary attacks. We prove that local variables of all normal agents can converge under the proposed algorithm if the trusted agents induce the connected dominating set of the original network. Furthermore, we exploit that the final solution of normal agents will converge to the convex optima set of the weighted average of all normal agents' local functions. We also show that the amount of tolerable adversary agents is not limited by the network connectivity under the proposed algorithm. Numerical results demonstrate the effectiveness of the proposed algorithm.**

I. INTRODUCTION

With high robustness and strong scalability, distributed optimization has found increasing applications in networked systems and Cyber Physical Systems (CPS), such as distributed machine learning [1], distributed time synchronization [2], distributed energy management [3], etc. Much attention has been paid to the design of distributed algorithms and the convergence analysis, which are applicable for different scenarios. For example, the dual averaging-based algorithm was proposed for the undirected or fixed communication topologies [4], [5], and the subgradient methods were developed for the directed time-varying graphs [6], and so on. However, most of existing works are built on the hypothesis that the network is deployed in benign surroundings without any intruder.

With the high integration of communication structure and distributed computation, the cyberattack has become one of the most challenging threats faced by CPS [7]. Under cyberattacks, the existing distributed optimization algorithms become vulnerable or even invalid, which may lead to the system paralysis [8]. Therefore, investigations on the resilient distributed optimization against attacks are desirable. Adversary attacks considered in the distributed optimization can be divided into malicious and Byzantine attacks [9]. Malicious attacks are intent to disrupt the network functions which can only send the same arbitrary state to their neighbors for each period information exchanging, while Byzantine attacks have the extra capability of sending different arbitrary states to different neighbors.

Recently, distributed optimization under adversary attacks has brought increasing interests. As a special case of distributed optimization, consensus under adversary agents has been studied for decades. Denote the maximum amount of tolerable attacks by F. Investigations on the relationship of F, the number of total normal agents and the network connectivity[1] have been provided for consensus under adversary attacks in [10] and the resilient algorithms were developed in [11], [12]. For the general distributed optimization under malicious attacks, Sundaram *et al.* proposed the resilient algorithm by removing F largest and smallest states at each iteration, under which the final solution will converge to the convex hull of the set of the all normal agents' local minimizers [8]. Then, considering the complete graph and Byzantine attacks, Su and Vaidya proposed a series of the resilient algorithms by removing F largest and smallest states and analyzed the convergence and optimality in [13], [14]. Then, in [15], [16], such results were extended to the constrained distributed optimization for directed networks and both of crash faults and Byzantine attacks were taken into consideration. It should be pointed out that all these algorithms are only effective when F is known, which is tightly associated with the connectivity of the network. To relax such limitation, Sundaram *et al.* proposed the local filtering consensus-based optimization in [9], [17] to bear F local adversary attacks, where the network has to be $(F+1, F+1)$-robust. Nonetheless, the effectiveness of the algorithm is still restricted by the network connectivity, meaning that when the network is sparse, the algorithm becomes invalid. Therefore, all the above existing resilient distributed optimization protocols will fail if F is not known or the network connectivity is very small.

To solve the above problem, we design a resilient distributed optimization algorithm by exploiting trusted agents (RDO-T), which is inspired by [18]. Abbas *et al.* in [18] proposed the resilient consensus with trusted agents (RCP-

1: The State Key Lab. of Industrial Control Technology and Innovation Joint Research Center for Industrial Cyber Physical Systems, Zhejiang University, China zccsq90@gmail.com;

2: Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China jianpinghe.zju@gmail.com;

3: Distinguished Professor, Institute of Intelligent Systems, University of Johannesburg, Johannesburg, South Africa wangq@uj.ac.za

This work was supported in part by NSFC under grant 61503332, U1509215, National Key R&D Program Under Grant 2016YFB0800204, and the Fundamental Research Funds for the Central Universities. Qing-Guo WANG acknowledges the support of South Africa NRF INCENTIVE FUNDING FOR RATED RESEARCHERS (grant No:109560), which partially funded his research on this work.

[1]The network connectivity means the least number of agents that can be cut from the network to make the rest network unconnected.

T), under which F can be any large number and the final state converges to the range of the maximum and minimum initial state. Different from [18], we consider the distributed optimization under adversary attacks. The main contributions of this work are summarized as follows:

1) We consider the problem of distributed optimization under adversary attacks, where the number of attack agents is not limited by the network connectivity and can be arbitrarily large.
2) We design RDO-T, which is the resilient distributed optimization by exploiting trusted agents against the adversary attacks. Under RDO-T, the normal agents only use the state bounded by the maximum and minimum state among its neighbors and itself for iterations.
3) We prove that under RDO-T, if the trusted agents induce a connected dominating set, the convergence can be achieved and the final solution can be bounded by the convex optima set of the weighted average of all normal agents' local functions.

The remainder of this paper is organized as follows. Section II provides the models and formulates the problem, before the detailed resilient distributed optimization algorithm is presented in Section III. Section IV analyzes the performance of the proposed algorithm. Section V tests the main results through numerical results. Conclusion is given in Section VI.

II. Preliminaries and Problem Formulation

A. Network Model

Consider a network with $n, n \geq 3$ agents with unique identifications (ID), which are denoted by $\{1, \cdots, n\}$. An undirected connected graph $G = \{V, E\}$ is used to represent the communication topology of the network, where V is the set of N agents and $E \subset V \times V$ is the edge set. As the network is undirected, there holds that $(j, i) \in E \Leftrightarrow (i, j) \in E$. We can also note that $(j, i) \in E$, if and only if (iff) agent i can receive information from agent j, and agent j is the neighbor of agent i. The neighbor set of agent i is defined as $N_i = \{j | (j, i) \in E, j \neq i\}$ and $|N_i|$ is the cardinality. Self-loop is not considered here, i.e., $(i, i) \notin E, \forall i \in V$. With a slight abuse of terminology, the terms agents and nodes will be utilized interchangeably.

We divide agents into three types, i.e., the normal agent, the trusted agent and the adversary agent.

1) **The normal agents** may crash or compromised by the attacker and they can identify the trusted agents among its neighbors. We denote the set of normal agents by V_n and its cardinality by $n_1 = |V_n|$.
2) **The trusted agents** are normal nodes which have a higher security level, implying that they cannot crash or be compromised by the attacker. The trusted agent can also identify the trusted agents in its neighbors. The trusted agent set is represented by V_t and the its cardinality is $n_2 = |V_t|$.
3) **The adversary agents** include malicious or Byzantine attacks. The adversary agents are aware of the network structure and can identify the trusted nodes. Meanwhile, the adversary agents know the update rule of normal agents. The set of adversary nodes is indicated by V_a and its cardinality is $n_a = |V_a|$.

Then, we provide the definition of the connected dominating set by referring to [19], which is useful in our analysis. And the assumption satisfied by the trusted nodes is also given.

Definition 2.1: A set D of graph $G = (V, E)$ is a connected dominating set if every node not belonging to D has at least one neighbor in D and all nodes in D form a connected graph.

Assumption 2.2: The set of trusted agents induce a connected dominating set of the graph $G = (V, E)$.

B. Distributed Optimization under Adversary Attacks

Assume that each agent i has its local cost function $f_i(x) \in R$, where $x \in R$ is the same variable owned by all agents. The local cost function $f_i(x), \forall i \in V$ is a convex, continuously differentiable function and is also Lipschitz continuous, i.e., $f_i'(x) \leq L$, where L is a constant. At the same time, it is supposed that the optimal point set $\arg\min f_i(x)$ is nonempty, bounded and closed. The derivative of the function $f_i(x), \forall i \in V$ is represented by $f_i'(x)$. The optimization problem can be written as,

$$\min \frac{1}{n} \sum_{i=1}^{n} f_i(x). \tag{1}$$

When adversary agents exist, for all normal nodes, the problem becomes,

$$\min \frac{1}{n_0} \sum_{i \in V_n \cup V_t} f_i(x), \tag{2}$$

where $n_0 = n_1 + n_2$. It is proved in [13] that if V_a is not an empty set, the problem (2) cannot be solved exactly in a distributed way.

C. Problem of Interests

Existing resilient distributed optimization algorithms, where the maximum number of tolerable attack nodes F is required, are in front of the following challenges:

- Each normal agent has to know the knowledge of F in some form, implying that all agents have to know the global information of the network to some extent.
- Since F has strong relationship with the network connectivity, when the connectivity of the network is very small, the existing algorithms become vulnerable. For example, when the connectivity of the network is 1, F is zero, meaning that the existing protocol is invalid as long as adversary attacks occur.

Therefore, it is desirable to design a resilient algorithm for distributed optimization under adversary attacks, under which F can be a arbitrarily number. Besides, how to guarantee the convergence of the local variable and how to evaluate the optimality of the proposed algorithm are also interesting. Inspired by the existing work in [18], we realize the resilient distributed optimization under adversary agents by enhancing the security of some subset of agents

in the network. The differences between our work and the work [18] involve: 1) RCP-T is only effective for consensus. Due to existence of cost functions, the convergence analysis for RCP-T is not suitable for distributed optimization under adversary attacks; 2) Authors provided the necessary and sufficient condition to guarantee the final state bounded by the maximum and minimum initial state of all normal agents [18]. But for distributed optimization, the objective is to make all local variables converge to the optimal solution. As the optimal solution of problem (2) cannot be obtained exactly, how to evaluate the optimality of the final state of the resilient algorithm against attacks will be more complex and challenging.

III. Algorithm Design

We provide the resilient distributed optimization algorithm by entrusting some subset of agents in this section. By protecting the connected dominating set of graph G, we can ensure that each normal node will only use neighbor's states which are bounded by the minimum and maximum one among neighboring trusted nodes and itself. Thus, the resilience can be guaranteed for the distributed optimization under adversary attacks even when the amount of attack nodes is large. Details of the resilient distributed optimization with trusted nodes (RDO-T) can be found in Algorithm 1.

Before giving the details, we provide some notations. In the iteration process of distributed optimization, the derivative of the local function of agent i at each iteration is denoted by $f_i'(k) = f_i'(x_i(k))$. Let $\{\alpha_0, \cdots .\alpha_\infty\}$ be the sequence of stepsizes, which satisfies $\sum_{k=0}^{\infty} \alpha_k = \infty$, $\sum_{k=0}^{\infty} \alpha_k^2 < \infty$ and $\alpha_{k+1} \leq \alpha_k$.

Algorithm 1 RDO-T

1. Each node i initializes $x_i(0)$ randomly and broadcasts the message $x_i(0)$ to its neighbors.
2. After receiving the message from neighbors, each node i identifies all neighboring trusted nodes, which are denoted by the set T_i. Then, node i sorts $x_j(k)$ for all $j \in T_i$ and keeps the maximum and minimum one as $x_{j_M}(k)$ and $x_{j_m}(k)$, respectively.
3. Node i compares $x_i(k)$ with $x_{j_M}(k)$ and $x_{j_m}(k)$, and then obtains the maximum and minimum state represented by $x_i^M(k)$ and $x_i^m(k)$. Then, by the comparison, node i obtains the set $R_i(k)$, i.e.,
$$R_i(k) = \{j | x_i^m(k) \leq x_j(k) \leq x_i^M(k), j \in N_i \cup i\}. \quad (3)$$
4. Each node i updates $x_i(k+1)$ according to (4). Then, node i broadcasts $x_i(k+1)$ to its neighbors.
$$x_i(k+1) = \frac{1}{|R_i(k)|} \sum_{j \in R_i(k)} x_j(k+1) - \alpha_k f_i'(k), \quad (4)$$
where $|R_i(k)|$ denotes the cardinality of $R_i(k)$.

IV. Performance Analysis

To evaluate the convergence and optimality performance of RDO-T, we first prove the existence of the transition matrix for all normal nodes at each iteration. Then the backward product of the transition matrices is given and by analyzing its ergodicity, we prove that the convergence can be achieved. Meanwhile, the convex optima set of the weighted average of all normal agents' functions by referring to [13] is given and we also show that the final solution of RDO-T will converge to this set.

A. Existence of Transition Matrix

Assume that all normal agents including trusted ones have identity $\{1, \cdots, n_0\}$. Then, we denote the state vector of all normal and trusted agents by $\mathbf{x}(k) = [x_1(k), \cdots, x_b(k)]^{\mathrm{T}} \in R^{n_0}$ at iteration k.

Lemma 4.1: If assumption 2.2 holds, under RDO-T, there exists $M(k)$, such that,
$$\mathbf{x}(k+1) = M(k)\mathbf{x}(k) - \alpha_k f'(k), \forall k, \quad (5)$$
where $f'(k) = [f'_1(k), \cdots, f'_{n_0}(k)]^{\mathrm{T}}$ and $M(k) = [M_{ij}(k)]_{n_0 \times n_0}$ satisfying

i) $M(k)$ is a row stochastic matrix, i.e.,
$$\sum_{j=1}^{n_0} M_{ij}(k) = 1, \forall i \in \{1, \cdots, n_0\}. \quad (6)$$

ii) $M_{ij}(k)$ is non-zero if and only if $(j,i) \in E$ or $j = i$.

iii) If $M_{ij}(k)$ is non-zero, there holds $M_{ij}(k) \geq \varphi = \frac{1}{d_M+1}$, where d_M the maximum cardinality of the neighboring set among all normal nodes.

Proof: Due to step 4 to 6 in RDO-T, each normal node i only utilizes the states $x_j(k), j \in R_i(k)$ of neighbors which are bounded by the maximum and minimum state among node i and its trusted neighbors. Hence, $R_i(k) \subseteq N_i$, that is, $\frac{1}{|R_i(k)|} \geq \frac{1}{|N_i|+1}$. Then, we analyze the following two cases, i.e., $R_i(k) \cap V_a = \emptyset$ and $R_i(k) \cap V_a \neq \emptyset$.

Case 1: If $R_i(k) \cap V_a = \emptyset$, then there holds $M_{ij}(k) = \frac{1}{|R_i(k)|}$ for $(j,i) \in E$ or $j = i$. Since all trusted agents induce a connected dominating set of graph G, Lemma 4.1 holds.

Case 2: If $R_i(k) \cap V_a \neq \emptyset$, there exists at least one adversary node in $R_i(k)$. Here, we consider that there is only one adversary node $s \in V_a$ in $R_i(k)$. Because of step 2 and 3 in Algorithm 1, there must exist $0 \leq \rho \leq 1$ such that $x_s(k) = \rho x_i^m(k) + (1-\rho)x_i^M(k)$. Based on (4), we have
$$\begin{aligned} & \frac{1}{|R_i(k)|} \sum_{j \in R_i(k)} x_j(k) \\ = & \frac{1}{|R_i(k)|} \Big(\sum_{j \in R_i(k) \backslash V_a} x_j(k) + x_s(k) \Big) \\ = & \frac{1}{|R_i(k)|} \Big(\sum_{j \in R_i(k) \backslash V_a} x_j(k) + \rho x_i^m(k) + (1-\rho) x_i^M(k) \Big), \end{aligned} \quad (7)$$
where $j \in R_i(k) \backslash V_a$ means that node j belongs to $R_i(k)$ but not V_a. Then, we consider there is only one node j_1 (j_2) in $R_i(k)$ such that $x_{j_1}(k) = x_i^m(k)$ ($x_{j_2}(k) = x_i^M(k)$) and give $R_i^1(k) = R_i(k) \backslash \{V_a \cup j_1 \cup j_2\}$. As $x_i^m(k) \in x_j(k), j \in R_i(k)$ and $x_i^M(k) \in x_j(k), j \in R_i(k)$, it obtains from (7) that
$$\begin{aligned} & \frac{1}{|R_i(k)|} \Big(\sum_{j \in R_i(k) \backslash V_a} x_j(k) + \rho x_i^m(k) + (1-\rho) x_i^M(k) \Big) \\ = & \frac{1}{|R_i(k)|} \Big(\sum_{j \in R_i^1(k)} x_j(k) + (1+\rho) x_{j_1}(k) + (2-\rho) x_{j_2}(k) \Big). \end{aligned}$$

From the above equation, we infer that $M_{ij}(k) = \frac{(1+\rho)}{|R_i(k)|}, j = j_1$, $M_{ij}(k) = \frac{(2-\rho)}{|R_i(k)|}, j = j_2$ and $M_{ij}(k) = \frac{(2-\rho)}{|R_i(k)|}, j \in R_i^1(k)$, which means that ii) and iii) hold. Since $|R_i^1(k)| + (1+\rho) + (2-\rho) = |R_i(k)|$, $M(k)$ is a row stochastic matrix, i) holds.

Similarly, we can obtain the same result for more than one adversary node and more than one node with maximum and minimum state in $R_i(k)$. ■

B. Backward Product and Its Ergodicity

Based on (5), it can be obtained,

$$\begin{aligned} & \mathbf{x}(k+1) \\ = \; & M(k)\mathbf{x}(k) - \alpha_k f'(k) \\ = \; & M(k)[M(k-1)\mathbf{x}(k-1) - \alpha_{k-1} f'(k-1)] - \alpha_k f'(k) \\ = \; & M(k)\cdots M(0)\mathbf{x}(0) - \sum_{t=0}^{k} (M(k)\cdots M(t+1))\alpha_t f'(t) \\ = \; & \Phi(k,0)\mathbf{x}(0) - \sum_{t=1}^{k+1} \Phi(k,t)\alpha_{t-1} f'(t-1), \end{aligned} \tag{8}$$

where $\Phi(k,t), t \le k+1$, is the backward product with $\Phi(k,k) = M(k)$ and $\Phi(k,k+1) = \mathrm{I}_{n_0}$. I_{n_0} is an identity matrix of size $n_0 \times n_0$. By referring the theories in [20] and [14], [16], we can obtain the following lemmas directly.

Lemma 4.2: If assumption 2.2 holds, under RDO-T, there are at least n_2 columns in $\Phi(t+n_0-1,t)$ lower bounded by $\varphi^{n_0}\mathbf{1}$ component-wise for all t, where $\mathbf{1} \in R^{n_0}$, is a vector with all elements equal to 1.

Lemma 4.3: If assumption 2.2 holds, under RDO-T, for $\Phi(k,t)$, there holds $\lim_{k\ge t, k\to\infty} \Phi(k,t) = \mathbf{1}\psi^{\mathrm{T}}(t)$, where $\psi(t)$ is a stochastic vector dependent on t.

Lemma 4.4: If assumption 2.2 holds, under RDO-T, for any $\Phi(k,t)$, we have $|\Phi_{ij}(k,t) - \psi_i(t)| \le (1-\varphi^{n_0})^{\lceil \frac{k-t+1}{n_0} \rceil}$.

Lemma 4.5: If assumption 2.2 holds, for any fixed t, at least n_2 entries in $\psi(t)$ are lower bounded by $(\frac{1}{n_0})^{n_0}$, i.e., there are at least n_2 entries $i \in \{1,\cdots,n_0\}$ such that,

$$\psi_i(t) \ge (\frac{1}{n_0})^{n_0}. \tag{9}$$

C. The Convex Optimal Set $\mathcal{C}(\mu,\nu)$

To evaluate the optimality of RDO-T, here we give the collection of functions and then prove the convexity of the collection by referring to [13]. Since the proposed algorithm in this paper is totally different, the bounds of the parameters μ and ν are different. The collection is denoted by,

$$\begin{aligned} \mathcal{C}(\mu,\nu) \; = \; & \{g(x) | g(x) = \sum_{i\in V_n\cup V_t} \beta_i f_i(x), \beta_i \ge 0, \\ & \sum_{i\in V_n\cup V_t} \beta_i = 1, \sum_{i\in V_n\cup V_t} \mathbf{1}\{\beta_i \ge \mu\} \ge \nu\} \end{aligned} \tag{10}$$

where $\mathbf{1}\{\cdot\}$ is the indication function. For any given μ and ν, $\mathcal{C}(\mu,\nu)$ is a valid function. Then, we define

$$Y(\mu,\nu) = \cup_{g(x)\in\mathcal{C}(\mu,\nu)} \arg\min_{x\in R} g(x). \tag{11}$$

Lemma 4.6: If $\mu \le (\frac{1}{n_0})^{n_0}, \nu \le n_2$, then $Y(\mu,\nu)$ is a convex set.

Proof: We provide $g_1(x) = \sum_{i\in V_n\cup V_t} a_i f_i(x)$, and $g_2(x) = \sum_{i\in V_n\cup V_t} b_i f_i(x)$, where $g_1(x), g_2(x), \in \mathcal{C}_(\mu,\nu)$ with $\mu \le (\frac{1}{n_0})^{n_0}, \nu \le n_2$. And we denote $x_1 \in \arg\min g_1(x)$, $x_2 \in \arg\min g_2(x)$ and consider $x_1 \ne x_2$. Given $g^*(x) = \sum_{i\in V_n\cup V_t} \frac{1}{n_0} f_i(x)$, since $\frac{1}{n_0} \ge (\frac{1}{n_0})^{n_0}$, there holds $g^*(x) \in \mathcal{C}_(\mu,\nu)$ with $\mu \le (\frac{1}{n_0})^{n_0}, \nu \le n_2$. It is also denoted that $x^* = \arg\min g^*(x)$. As a result, $x^* \in Y(\mu,\nu)$.

Considering that $x_3 = \varrho x_1 + (1-\varrho)x_2$, where $0 \le \varrho \le 1$, we will prove the convexity of the set $Y(\mu,\nu)$ from the following two cases based on [13]:

Case 1: $x_3 \in \arg\min g_1(x) \cup \arg\min g_2(x) \cup \arg\min g^*(x)$;

Case 2: $x_3 \notin \arg\min g_1(x) \cup \arg\min g_2(x) \cup \arg\min g^*(x)$.

We first consider case 1. Due to $x_3 \in \arg\min g_1(x) \cup \arg\min g_2(x) \cup \arg\min g^*(x)$, we have $x_3 \in Y(\mu,\nu)$, meaning that $Y(\mu,\nu)$ is a convex set.

For case 2, as $x_3 \notin \arg\min g_1(x) \cup \arg\min g_2(x)$, there must hold that $x_1 \ne x_3 \ne x_2$. Considering $x_1 < x_2$, we have $x_1 < x_3 < x_2$. Since $x_1 = \arg\min g_1(x)$ and $x_2 = \arg\min g_2(x)$, we obtain that $x_3 > \max[\arg\min g_1(x)]$ and $x_3 < \min[\arg\min g_2(x)]$, i.e., ${g'}_1(x_3) > 0$ and ${g'}_2(x_3) < 0$. Meanwhile, because $x_3 \notin \arg\min g^*(x)$, we must have ${g^*}'(x_3) > 0$ or ${g^*}'(x_3) < 0$. Here, we consider that ${g^*}'(x_3) > 0$. For the case ${g^*}'(x_3) < 0$, the following result can be obtained for the same reason.

When ${g^*}'(x_3) > 0$, there must exist $0 < \sigma < 1$ such that $\sigma {g^*}'(x_3) + (1-\sigma){g'}_2(x_3) = 0$. Then, we infer that

$$\sigma \frac{1}{n_0} \sum_{i\in V_n\cup V_t} f_i'(x_3) + (1-\sigma) \sum_{i\in V_n\cup V_t} b_i f_i'(x_3) = 0. \tag{12}$$

Hence,

$$\sum_{i\in V_n\cup V_t} (\sigma\frac{1}{n_0} + (1-\sigma)b_i) f_i'(x_3) = 0, \tag{13}$$

where $\sum_{i\in V_n\cup V_t} (\sigma\frac{1}{n_0} + (1-\sigma)b_i) = 1$. From (13), we obtain that $x_3 \in \arg\min g_3(x) = \sum_{i\in V_n\cup V_t} (\sigma\frac{1}{n_0} + (1-\sigma)b_i) f_i(x)$. Then we will prove that $g_3(x) \in \mathcal{C}(\mu,\nu)$ with $\mu \le (\frac{1}{n_0})^{n_0}, \nu \le n_2$, which indicates that $x_3 \in Y(\mu,\nu)$.

As $g_2(x) \in \mathcal{C}(\mu,\nu)$, there are at least ν elements b_i satisfying $b_i \ge \mu \ge (\frac{1}{n_0})^{n_0}$. Let $S_1 = \{b_i | b_i \ge \mu \ge (\frac{1}{n_0})^{n_0}\}$. For each $i \in S_i$, due to $\frac{1}{n_0} \ge \frac{1}{n_0})^{n_0}$, we have

$$\sigma\frac{1}{n_0} + (1-\sigma)b_i \ge \sigma(\frac{1}{n_0})^{n_0} + (1-\sigma)(\frac{1}{n_0})^{n_0} \ge (\frac{1}{n_0})^{n_0}. \tag{14}$$

Therefore, there are at least ν weights $(\sigma\frac{1}{n_0} + (1-\sigma)b_i)$ satisfying $(\sigma\frac{1}{n_0} + (1-\sigma)b_i) \ge \mu$, implying $x_3 \in Y(\mu,\nu)$. Hence, $Y(\mu,\nu)$ is a convex set if $\mu \le (\frac{1}{n_0})^{n_0}, \nu \le n_2$. ■

D. Convergence and Optimality of RDO-T

Based on the above results, we prove that the convergence of RDO-T can be guaranteed and the final solution will always converge to the optima set $Y(\mu,\nu)$. The main idea of the proof has referred to the theoretical results in [16],

but we claim that the proposed algorithm and parameters are different so that the proof is different.

Theorem 4.7: If assumption 2.2 holds, $\mu \leq (\frac{1}{n_0})^{n_0}, \nu \leq n_2$ and $\lim_{k\to\infty} \alpha_k = 0$, RDO-T will converge and the final solution will belong to $Y(\mu,\nu)$, i.e.,

$$\lim_{k\to\infty} x_i(k) = \lim_{k\to\infty} x_j(k) \in Y(\mu,\nu), \forall i,j \in V_n \cup V_t. \quad (15)$$

Proof: Assume that after iteration $\bar{k}$, all agents will set $f_i(k) = 0, k \geq \bar{k}$. Then, there exists

$$\begin{aligned}
& \mathbf{x}(k+\bar{k}) \\
= \;& \Phi(k,\bar{k}-1)\mathbf{x}(\bar{k}-1) - \sum_{t=1}^{\bar{k}} \Phi(k,t)(\alpha_{t-1}f'(t-1)) \\
= \;& \Phi(k,\bar{k}-1)\mathbf{x}(\bar{k}-1) - \Phi(k,\bar{k}) \sum_{t=1}^{\bar{k}} \Phi(\bar{k}-1,t)(\alpha_{t-1}f'(t-1)) \\
= \;& \Phi(k,\bar{k})[\Phi(\bar{k},\bar{k}-1)\mathbf{x}(\bar{k}-1) - \sum_{t=1}^{\bar{k}} \Phi(\bar{k}-1,t)(\alpha_{t-1}f'(t-1))] \\
= \;& \Phi(k,\bar{k})\mathbf{x}(\bar{k}). \quad (16)
\end{aligned}$$

Taking limitations on both sides of (16), it can be obtained

$$\begin{aligned}
& \lim_{k\to\infty} \mathbf{x}(k+\bar{k}) \\
= \;& \lim_{k\to\infty} [\Phi(k,0)\mathbf{x}(0) - \sum_{t=1}^{\bar{k}} \Phi(k,t)(\alpha_{t-1}f'(t-1))] \\
= \;& \lim_{k\to\infty} \Phi(k,0)\mathbf{x}(0) - \sum_{t=1}^{\bar{k}} \lim_{k\to\infty} \Phi(k,t)(\alpha_{t-1}f'(t-1))] \\
= \;& \mathbf{1}\psi^{\mathrm{T}}(0)\mathbf{x}(0) - \sum_{t=1}^{\bar{k}} \alpha_{t-1}\mathbf{i}\psi^{\mathrm{T}}(t)f'(t-1) \\
= \;& [\langle \psi^{\mathrm{T}}(0), \mathbf{x}(0)\rangle - \sum_{t=1}^{\bar{k}} \alpha_{t-1}\langle \psi^{\mathrm{T}}(t), f'(t-1)\rangle]\mathbf{1}, \quad (17)
\end{aligned}$$

where $\langle , \rangle$ is the symbol for inner product of two vectors. We see that the limit of vector $\mathbf{x}(k+\bar{k})$ is a vector with all elements equal to one constant denoted by $y(\bar{k})$. Then, let $\mathbf{y}(\bar{k}) = [y(\bar{k}), \cdots, y(\bar{k})]^{\mathrm{T}} \in R^{n_0}$. According to (17), $y(\bar{k})$ is rewritten as

$$\begin{aligned}
y(\bar{k}) &= \langle \psi^{\mathrm{T}}(0), \mathbf{x}(0)\rangle - \sum_{t=1}^{\bar{k}} \alpha_{t-1}\langle \psi^{\mathrm{T}}(t), f'(t-1)\rangle \\
&= \langle \psi^{\mathrm{T}}(0), \mathbf{x}(0)\rangle - \sum_{t=1}^{\bar{k}-1} \alpha_{t-1}\langle \psi^{\mathrm{T}}(t), f'(t-1)\rangle \\
&\quad -\langle \alpha_{\bar{k}-1}\langle \psi^{\mathrm{T}}(\bar{k}), f'(\bar{k}-1)\rangle \\
&= y(\bar{k}-1) - \alpha_{\bar{k}-1}\langle \psi^{\mathrm{T}}(\bar{k}), f'(\bar{k}-1)\rangle. \quad (18)
\end{aligned}$$

Thus, $\lim_{k\to\infty} y(k+1)$ can be seen as the optimal solution of the function $\sum_{i=1}^{n_0} \psi_j(t-1)f_i(x)$. Based on Lemma 4.5, there are at least n_2 entries j such that $\psi_j(t-1) > (\frac{1}{n_0})^{n_0}$. Therefore,

$$\lim_{k\to\infty} y(k+1) \in Y(\mu,\nu), \quad (19)$$

where $\mu \leq (\frac{1}{n_0})^{n_0}, \nu \leq n_2$. Meanwhile, we note that

$$\begin{aligned}
y(k+1) &= \sum_{j=1}^{n_0} \psi_j(0)x_j(0) - \\
&\quad \sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \psi_j(t)(\alpha_{t-1}f'_j(t-1)). \quad (20)
\end{aligned}$$

For each node $i \in V_n \cup V_t$, $x_i(k+1)$ satisfies,

$$\begin{aligned}
x_i(k+1) &= \sum_{j=1}^{n_0} \Phi_{ij}(k,0)x_j(0) - \\
&\quad \sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \Phi_{ij}(k,t)(\alpha_{t-1}f'_j(t-1)). \quad (21)
\end{aligned}$$

Then, combining (18) and (20), we obtain that

$$\begin{aligned}
& |x_i(k+1) - y(k+1)| \\
= \;& |\sum_{j=1}^{n_0} \Phi_{ij}(k,0)x_j(0) - \sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \Phi_{ij}(k,t)(\alpha_{t-1}f'_j(t-1)) \\
& [\sum_{j=1}^{n_0} \psi_j(0)x_j(0) - \sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \psi_j(t)(\alpha_{t-1}f'_j(t-1))]| \\
\leq \;& |\sum_{j=1}^{n_0} \Phi_{ij}(k,0)x_j(0) - \sum_{j=1}^{n_0} \psi_j(0)x_j(0)| + \\
& |\sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \Phi_{ij}(k,t)(\alpha_{t-1}f'_j(t-1)) \\
& -\sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \psi_j(t)(\alpha_{t-1}f'_j(t-1))|. \quad (22)
\end{aligned}$$

On the one hand, we have,

$$\begin{aligned}
& |\sum_{j=1}^{n_0} \Phi_{ij}(k,0)x_j(0) - \sum_{j=1}^{n_0} \psi_j(0)x_j(0)| \\
= \;& |\sum_{j=1}^{n_0} (\Phi_{ij}(k,0) - \psi_j(0))x_j(0)| \\
\leq \;& n_0(1-\varphi^{n_0})^{\lceil \frac{k+1}{n_0} \rceil} \max[|x_i(0)|]. \quad (23)
\end{aligned}$$

On the other hand, since $f_i(x)$ is Lipschitz continuous,

$$\begin{aligned}
& |\sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \Phi_{ij}(k,t)(\alpha_{t-1}f'_j(t-1)) \\
& -\sum_{t=1}^{k+1}\sum_{j=1}^{n_0} \psi_j(t)(\alpha_{t-1}f'_j(t-1))| \\
= \;& |\sum_{t=1}^{k+1} (\Phi_{ij}(k,0) - \psi_j(0)(\alpha_{t-1}f'_j(t-1)))| \\
\leq \;& n_0 L \sum_{t=1}^{k+1} \alpha_{t-1}(1-\varphi^{n_0})^{\lceil \frac{k-t+1}{n_0} \rceil}. \quad (24)
\end{aligned}$$

Combining (23) and (24), it infers that

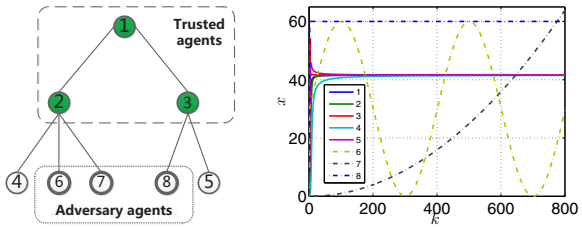
$$\begin{aligned}
& \lim_{k\to\infty} |x_i(k+1) - y(k+1)| \\
\leq \;& n_0 \lim_{k\to\infty} (1-\varphi^{n_0})^{\lceil \frac{k+1}{n_0} \rceil} \max[|x_i(0)|] + \\
& n_0 L \lim_{k\to\infty} \sum_{t=1}^{k+1} \alpha_{t-1}(1-\varphi^{n_0})^{\lceil \frac{k-t+1}{n_0} \rceil} \\
= \;& 0. \quad (25)
\end{aligned}$$

From (19) and (25), we have that $\forall i \in V_n \cup V_t$,

$$\lim_{k\to\infty} x_i(k+1) \in Y(\mu,\nu), \quad (26)$$

where $\mu \leq (\frac{1}{n_0})^{n_0}, \nu \leq n_2$. Thus, we conclude the result. ■

Remark 4.8: It can be seen that there is no requirement for the information regarding the number of tolerable attack agents in RDO-T and no condition for the number of the



(a) an example (b) convergence

Fig. 1. An network example and performance evaluation.

tolerable attack agents is needed to guarantee the convergence and optimality. Hence, RDO-T is effective even when the number of adversary attacks is very large. Therefore, the maximum amount of tolerable attack agents has no relationship with the network connectivity and when the network connectivity is small, RDO-T can still be resilient for the distributed optimization against attacks.

V. Numerical Results

In this section, we investigate the effectiveness of the proposed algorithm through numerical results. Consider a network with $n = 8$ agents including 3 adversary attacks, 3 trusted agents and 2 normal agents, which is shown in Fig. 1(a). Consider the local cost function of each agent $i, \forall i \in V$ has the expression shown in (27) with different parameters e_i, $p_i > 0$ and q_i selecting from intervals [8,20], [0,70], [-50 80], respectively.

$$f_i(x) = p_i\sqrt{e_i^2 + (x - q_i)^2} \tag{27}$$

For each iteration $k \geq 1$, the adversary agents will violate the rule in RDO-T and update their states as follows,

$$\begin{aligned} x_i &= 30sin(0.005\pi x_i(k-1) * k) + 30, i = 6, \\ x_i &= (k/100)^2, i = 7, \\ x_i &= 60, i = 8. \end{aligned} \tag{28}$$

From Fig. 1(b), we can see that all states of normal agents and trusted agents will converge under adversary attacks. Furthermore, as the connectivity of the network shown in Fig. 1(a) is 1, the existing algorithm in [16] cannot tolerate one attack node. Meanwhile, the protocol proposed in [9] will also be invalid since the network is not (2,2)-robust. Compared with these existing works, as long as the trusted agents induce the connected dominating set, RDO-T can be effective even when there are lots of adversary attacks. Besides that, it should be pointed out that when the network connectivity is large, the connected dominating set is very small, implying that we only need protect small amount of agents to enhance the resilience.

VI. Conclusion

In this paper, we developed the resilient distributed optimization algorithm under adversary attacks by making some connected dominating set of nodes have the high security. We proved that the convergence can be guaranteed and the final solution will definitely join in the optima set of some weighted average of all normal agents' functions. When the number of tolerable adversary attacks is not strictly limited by the network connectivity, RDO-T is still resilient for the distributed optimization. It means that when the connectivity of the network is small, the proposed algorithm is still effective. Furthermore, we found that when the connectivity of the network is large, we only need protect a small amount of agents with the high security for RDO-T.

References

[1] L. Su and N. Vaidya. Robust multi-agent optimization: coping with byzantine agents with input redundancy. In *Proc. Symposium on Stabilization, Safety, and Security of Distributed Systems*, pages 368–382. Springer, 2016.

[2] J. He, P. Cheng, L. Shi, and J. Chen. Time synchronization in wsns: A maximum value based consensus approach. In *Proc. IEEE CDC*, pages 7882–7887. IEEE, 2011.

[3] C. Zhao, J. He, P. Cheng, and J. Chen. Dual averaging for distributed optimization: convergence analysis and network scaling. *IEEE Transactions on Smart Grid*, 2016.

[4] A. Nedic and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

[5] J. C. Duchi, A. Agarwal, and M. J. Wainwright. Dual averaging for distributed optimization: convergence analysis and network scaling. *IEEE Transactions on Automatic Control*, 57(3):592–606, 2012.

[6] A. Nedic and A. Ozdaglar. Distributed optimization over time-varying directed graphs. *IEEE Transactions on Automatic Control*, 60(3):601–615, 2015.

[7] F. Pasqualetti, F. Drfler, and F. Bullo. Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11):2715–2729, 2013.

[8] S. Sundaram and B. Gharesifard. Consensus-based distributed optimization with malicious nodes. In *Proc. Allerton*, pages 244–249. IEEE, 2015.

[9] S. Sundaram and B. Gharesifard. Distributed optimization under adversarial nodes. *arXiv preprint arXiv:1606.08939*, 2016.

[10] M. Pease, R. Shostak, and L. Lamport. Reaching agreement in the presence of faults. *Journal of the ACM*, 27(2):228–234, 1980.

[11] F. Pasqualetti, A. Bicchi, and F. Bullo. Consensus computation in unreliable networks: A system theoretic approach. *IEEE Transactions on Automatic Control*, 57(1):90–104, 2012.

[12] H. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram. Resilient asymptotic consensus in robust networks. *IEEE Journal on Selected Areas in Communications*, 31(4):766–781, 2013.

[13] L. Su and N. Vaidya. Byzantine multi-agent optimization: Part i. *arXiv preprint arXiv:1506.04681*, 2015.

[14] L. Su and N. Vaidya. Byzantine multi-agent optimization: Part ii. *arXiv preprint arXiv:1507.01845*, 2015.

[15] L. Su and N. Vaidya. Fault-tolerant multi-agent optimization: Part iii. *arXiv preprint arXiv:1509.01864*, 2015.

[16] L. Su and N. Vaidya. Fault-tolerant distributed optimization (part iv): constrained optimization with arbitrary directed networks. *arXiv preprint arXiv:1511.01821*, 2015.

[17] S. Sundaram and B. Gharesifard. Secure local filtering algorithms for distributed optimization. In *Proc. CDC*, pages 1871–1876. IEEE, 2016.

[18] W. Abbas, Y. Vorobeychik, and X. Koutsoukos. Resilient consensus protocol in the presence of trusted node. In *Proc. ISRCS*, pages 1–7. IEEE, 2014.

[19] Y. Caro, D. B. West, and R. Yuster. Connected domination and spanning trees with many leaves. *SIAM Journal on Discrete Mathematics*, 13(2):202–211, 2000.

[20] N. Vaidya. Matrix representation of iterative approximate byzantine consensus in directed graphs. *arXiv preprint arXiv:1203.1888*, 2012.