Dependable Distributed Nonconvex Optimization via Polynomial Approximation

Zhiyu He, Jianping He, Cailian Chen and Xinping Guan

***Abstract*—There has been work on exploiting polynomial approximation to solve distributed nonconvex optimization problems. This idea facilitates arbitrarily precise global optimization without requiring local evaluations of gradients at every iteration. Nonetheless, there remains a gap between existing theoretical guarantees and diverse practical requirements for dependability, including privacy preservation and robustness to network imperfections (e.g., time-varying directed communication, asynchrony and packet drops). To fill this gap and keep the above strengths, we propose a Dependable Chebyshev-Proxy-based distributed Optimization Algorithm (D-CPOA). Specifically, to ensure both accuracy of solutions and privacy preservation of local objective functions, a new privacy-preserving mechanism is designed. This mechanism leverages the randomness in block-wise insertions of perturbed data and separate subtractions of added noises, and its effects are thoroughly analyzed through (α, β)-data-privacy. In addition, to gain robustness to various network imperfections, we use the push-sum consensus protocol as a backbone, discuss its specific enhancements, and evaluate the performance of the proposed algorithm accordingly. Thanks to the linear consensus-based structure of iterations, we avoid the privacy-accuracy trade-off and the bother of selecting appropriate step-sizes in different settings. We provide rigorous treatments of the accuracy, dependability and complexity. It is shown that the advantages brought by the idea of polynomial approximation are perfectly maintained when all the above challenging requirements exist. Simulations demonstrate the efficacy of the developed algorithm.**

***Index Terms*—Distributed optimization, Chebyshev polynomial approximation, dependability, privacy preservation, (α, β)-data-privacy, robustness.**

I. Introduction

Distributed optimization enables multiple agents in a network to agree on the optimal points of the average of local objective functions. This global aim is achieved by exploiting local computations and communication between neighboring agents. Such a distributed architecture is highly preferable in a variety of applications related to large-scale networked systems, e.g., distributed learning [2], energy management [3] and resource allocation [4]. In these applications, the needs of improving efficiency, scalability and robustness as well as protecting privacy have motivated the development of distributed strategies, which serve as plausible alternatives to their centralized counterparts.

A. Motivations

Considerable effort has been devoted to designing efficient distributed optimization algorithms, e.g., [5]–[8] and extending them to meet diverse practical requirements, including privacy preservation [4], [9], [10], time-varying and directed communication [11]–[13] and asynchronous computations to allow lack of coordination [13], [14], delays and packet drops [15], [16]. Most of these extensions focus on convex problems, and some critical and complex issues including privacy-accuracy trade-off [10], optimal numbers of iterations [4] and bounds for constant step-sizes [13], [16] have been explored.

Recently, [17] proposed a promising algorithm termed CPCA to address a class of constrained distributed nonconvex optimization problems. The core idea is to first use polynomial approximations (i.e., proxies) to substitute for general local objective functions, and then employ consensus protocols, where the information of coefficients of these proxies is exchanged, to enable agents to acquire a global proxy, and finally solve an easier approximate version of the original problem locally. The novel idea of employing polynomial approximation helps to achieve arbitrarily precise global optimization without demanding local evaluations of gradients or values of objective functions at every iteration. More importantly, it separates this algorithm from typical gradient-based methods, and offers a new perspective to address distributed optimization problems.

Nevertheless, this algorithm is neither inherently privacy-preserving nor robust against various imperfections in network communication, and it is unclear whether the aforementioned advantages can be maintained when the factors of privacy and robustness are taken into account. First, it can easily cause the leakage of private information of local objective functions. The leakage results from its consensus-based iterations where the vectors of coefficients of local proxies are directly exchanged. Once the adversaries obtain the exact initial vector of the target agents, they can recover a fairly accurate estimate of the corresponding local objectives. Hence, how to effectively preserve the privacy of objective functions in this algorithm and to quantify the protection results are well worth consideration. Second, it only handles the optimization over static undirected graphs with perfect communication. Given that issues including time-varying and directed links, lack of coordination, transmission delays and packet drops are common in applications, it is meaningful to investigate their effects on the performance of this algorithm and find effective measures to make it more robust. The above issues have motivated the study of this work. We aim to demonstrate that the novel idea of introducing polynomial approximation into distributed

The authors are with the Department of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai 200240, China. Emails: {hzy970920, jphe, cailianchen, xpguan}@sjtu.edu.cn. This paper was presented in part at the 59th IEEE Conference on Decision and Control, Republic of Korea, December 2020 [1].

optimization not only allows for further enhancements to meet diverse practical needs, but also maintains notable advantages in terms of performance in these settings.

B. Contributions

In this paper, we exploit the idea of using polynomial approximation and develop a Dependable Chebyshev-Proxy-based distributed Optimization Algorithm (D-CPOA), considering typical needs of privacy preservation and robustness to various network imperfections, including time-varying and directed communication and asynchrony due to lack of coordination, delays or packet drops.

We first focus on the requirement of preserving the privacy of local objective functions. This requirement is reduced to keeping the initial vectors as secrets in the consensus-based iterations of CPCA. These vectors store the coefficients of local approximations and are of different lengths. Instead of simply extending existing methods designed for the scalar case (e.g., [18]–[21]), we exploit the feature that vector states can be partitioned and propose a new privacy-preserving mechanism for consensus-based iterations. The key idea is to append initial vectors perturbed by noises block by block to current states and then remove the influence of perturbations at several separate iterations afterward. The randomness in the actions of appending helps to hide useful perturbed initial values within iterations, and the separate subtractions of noises can ensure the exact convergence and also mitigate the negative impacts of persistent noises on the convergence rates [20]. These designs contribute to the effective preservation of privacy, and its degree is properly analyzed through (α, β)-data-privacy [22]. We avoid the trade-off between privacy and accuracy and consider a more general problem with nonconvex objectives, which is in sharp contrast with existing differentially private distributed convex optimization algorithms [4], [9], [10].

To gain robustness against various imperfections in network communication, we employ the push-sum average consensus protocol [23] as a backbone of iterations to handle time-varying and directed graphs, and then discuss its asynchronous extensions to cope with issues including lack of coordination, delays and packet drops. We analyze in detail the relationship between the accuracy of consensus and that of the obtained solutions, thus verifying that the proposed algorithm keeps effective when the above network imperfections exist. Since the iterations of the developed algorithm are linear and consensus-based, we are free from the problem of selecting appropriate step-sizes in different settings, which is a troublesome routine of typical gradient-based methods.

Preliminary results on addressing time-varying and directed communication and some computational issues are presented in [1]. In this paper, we extend the analysis by i) further fulfilling the requirement of privacy preservation, ii) offering more details on the design and analysis of the strategies to deal with various network imperfections, and iii) adding omitted proofs relating to the analysis of the developed algorithm. The main contributions are summarized as follows.

1) We propose D-CPOA to solve a class of constrained distributed nonconvex optimization problems, pursuing the guarantee of privacy preservation and robustness against various network imperfections. We demonstrate that it maintains the advantages of CPCA in being able to obtain ϵ-globally optimal solutions for any arbitrarily small given precision ϵ and being distributed terminable.
2) A new privacy-preserving mechanism is introduced into the proposed algorithm to prevent sensitive information of local objective functions from being leaked. This mechanism is tailored for the setting where local objective functions are represented as vectors of different lengths, and it exploits the randomness in block-by-block insertions of perturbed data and separate subtractions of added noises to achieve both the effective preservation of privacy and the exact convergence. We thoroughly analyze the privacy degree through (α, β)-data-privacy.
3) We address the robustness issue of the proposed algorithm in face of various imperfections in network communication. The iterations of D-CPOA are based on the push-sum consensus protocol. It functions well over time-varying and directed networks and can be further enhanced to allow for asynchronous computations and manage time delays and packet drops. We prove that the proposed algorithm keeps effective when all these imperfections are present, and there is no need to carefully select proper step-sizes in different circumstances.

C. Paper Organization

The remainder of this paper is organized as follows. Section II describes the problem of interest and gives some preliminaries. Section III presents the algorithm D-CPOA. Section IV analyzes the accuracy, dependability and complexity of the proposed algorithm. Numerical evaluations are performed in Section V, followed by the review of related work in Section VI. Finally, Section VII concludes this paper.

II. Problem Description and Preliminaries

Consider a network system consisting of N agents, each of which owns a local objective function $f_i(x) : X_i \to \mathbb{R}$ and a local constraint set $X_i \subset \mathbb{R}$. The network at time $t (t \in \mathbb{N})$ is described as a directed graph $\mathcal{G}^t = (\mathcal{V}, \mathcal{E}^t)$, where $\mathcal{V}$ is the set of agents, and $\mathcal{E}^t \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. Note that $(i, j) \in \mathcal{E}^t$ if and only if (*iff*) agent j can receive messages from agent i at time t. In this paper, the superscript t, subscripts i, j and script in parentheses k denote the number of iterations, indexes of agents and index of elements in a vector, respectively.

A. Problem Description

In this paper, we aim to solve the following constrained optimization problem

$$\begin{aligned} \min_{x} \quad & f(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x), \\ \text{s.t.} \quad & x \in X = \bigcap_{i=1}^{N} X_i \end{aligned} \tag{1}$$

in a distributed and dependable manner. Specifically, the global aim of optimization needs to be achieved by means of local communication and computations. Meanwhile, diverse practical requirements will be taken into account, including preservation of the privacy of local objective functions and robustness to time-varying directed communication and asynchrony. Some basic assumptions are given as follows.

Assumption 1. *Every $f_i(x)$ is Lipschitz continuous on X_i.*

Assumption 2. *All X_i are closed, bounded and convex sets.*

Assumption 3. *$\{\mathcal{G}^t\}$ is B-strongly-connected, i.e., there exists a positive integer B, such that for any $k \in \mathbb{N}$, the graph $\big(\mathcal{V}, \bigcup_{t=kB}^{(k+1)B-1} \mathcal{E}^t\big)$ is strongly connected.*

In problem (1), the objective functions are (possibly) nonconvex and the constraint sets are convex. Therefore, it is a constrained distributed nonconvex optimization problem. Under Assumption 2, X_i is a closed interval for any $i \in \mathcal{V}$. Hence, let $X_i = [a_i, b_i]$, where $a_i, b_i \in \mathbb{R}$. As a result, $X = [a, b]$, where $a = \max_{i\in\mathcal{V}} a_i,\ b = \min_{i\in\mathcal{V}} b_i$.

B. Preliminaries

• *Consensus Protocols*

Let $\mathcal{N}_i^{\text{in},t} = \{j|(j,i) \in \mathcal{E}^t\}$ and $\mathcal{N}_i^{\text{out},t} = \{j|(i,j) \in \mathcal{E}^t\}$ be the sets of agent i's in-neighbors[1] and out-neighbors, respectively, and $d_i^{\text{out},t} = |\mathcal{N}_i^{\text{out},t}|$ be its out-degree, where $|\mathcal{N}_i^{\text{out},t}|$ is the cardinality of $\mathcal{N}_i^{\text{out},t}$. Suppose that every agent i owns a local variable $x_i^t \in \mathbb{R}$. There are two kinds of classical consensus protocols, i.e., maximum consensus and average consensus, that allow agents to reach a global agreement through local information exchange only. The maximum consensus protocol [25] is given by

$$x_i^{t+1} = \max_{j\in\mathcal{N}_i^{\text{in},t}} x_j^t. \tag{2}$$

It can be proven that with (2), all x_i^t converge to $\max_{i\in\mathcal{V}} x_i^0$ in $T(\le (N-1)B)$ iterations, i.e,

$$x_i^t = \max_{i\in\mathcal{V}} x_i^0, \quad \forall t \ge T,\ i \in \mathcal{V}.$$

The push-sum average consensus protocol [23] is given by

$$x_i^{t+1} = \sum_{j\in\mathcal{N}_i^{\text{in},t}} a_{ij}^t x_j^t, \quad y_i^{t+1} = \sum_{j\in\mathcal{N}_i^{\text{in},t}} a_{ij}^t y_j^t, \tag{3}$$

where $y_i^t \in \mathbb{R}$ is initialized to be 1 for all $i \in \mathcal{V}$. The weight a_{ij}^t is set according to

$$a_{ij}^t = \begin{cases} 1/d_j^{\text{out},t}, & \text{if } j \in \mathcal{N}_i^{\text{in},t}, \\ 0, & \text{else.} \end{cases} \tag{4}$$

Let $A^t \triangleq (a_{ij}^t)_{N\times N}$. It follows that A^t is column stochastic. In the implementation, (3) requires every agent j to transmit the data $x_j^t/d_j^{\text{out},t}$ and $y_j^t/d_j^{\text{out},t}$ to its out-neighbors. With (3), the ratio $z_i^t \triangleq x_i^t/y_i^t$ converges geometrically to the average of all the initial values $\bar{x} = 1/N \sum_{i=1}^N x_i^0$ [23], i.e.,

$$\lim_{t\to\infty} z_i^t = \bar{x}, \quad \forall i \in \mathcal{V}.$$

• *Chebyshev Polynomial Approximation*

The degree m Chebyshev interpolant $p^{(m)}(x)$ corresponding to a Lipschitz continuous function $g(x)$ defined on $[a, b]$ takes the form as

$$p^{(m)}(x) = \sum_{j=0}^{m} c_j T_j\Big(\frac{2x-(a+b)}{b-a}\Big), \quad x \in [a,b], \tag{5}$$

where $T_j(u)$ is the j-th Chebyshev polynomial defined on $[-1, 1]$ and satisfies $|T_j(u)| \le 1,\ \forall u \in [-1, 1]$. As m increases, $p^{(m)}(x)$ uniformly converges to $g(x)$ on the entire $[a, b]$ [26], i.e.,

$$\forall x \in [a,b],\ \big|p^{(m)}(x) - g(x)\big| \to 0, \text{ as } m \to \infty.$$

Note that the convergence rates of approximation errors depend on the smoothness of $g(x)$ and are discussed in [17]. Consequently, computing $p^{(m)}(x)$ becomes a practical way to construct an arbitrarily precise polynomial approximation for $g(x)$, as theoretically ensured by the *Weierstrass Approximation Theorem* [26].

C. Models of Adversaries of Privacy

In this paper, we mainly consider the *honest-but-curious adversaries* [21]. These adversaries are agents in the network that faithfully follow the specified protocol but intend to infer $f_i(x)$ of the target agent i based on the received data.

In applications, the evolutions of time-varying networks can be arbitrary and unpredictable. Hence, it is hard for these adversaries to own stable and perfect knowledge of the key information on which an accurate estimation relies. In this paper, we are first concerned with the issue of privacy disclosure arising in the consensus iterations of D-CPOA. As for the push-sum consensus algorithms, this key information available to each agent i refers to

$$I_i^{\text{own},t} = \{a_{ii}^t, x_i^t\}, \quad I_i^{\text{in},t} = \{a_{ij}^t, x_j^t | j \in \mathcal{N}_i^{\text{in},t}\},$$

which are information sets of the states and weights of agent i and those transmitted from $\mathcal{N}_i^{\text{in},t}$ to agent i at time t, respectively. As has been discussed in [18], [27], the knowledge of $\bigcup_{t\in\mathbb{N}} I_i^{\text{own},t}$, $\bigcup_{t\in\mathbb{N}} I_i^{\text{in},t}$ and the coupling relationship between the locally added noises is a sufficient condition for the privacy compromise of the noise-adding-based privacy-preserving consensus algorithms. We make the following assumption on the abilities of these adversaries.

Assumption 4. *At every time t, for the target agent i, honest-but-curious adversaries can always access $I_i^{own,t}$ but can only obtain the full knowledge of $I_i^{in,t}$ with a probability whose upper bound is $p \in (0, 1)$.*

D. Privacy Definition

Without loss of generality, we consider the requirement of preserving the privacy of agent i's local objective $f_i(x)$. In CPCA, local communication happens in its second stage of average consensus iterations, where agents directly exchange and update their local variables $p_i^0 \in \mathbb{R}^{m_i+1}$. These variables are the vectors of coefficients of approximations $p_i(x)$ for local objectives $f_i(x)$. Once the adversaries obtain an estimation $\hat{p}_i$

[1] As [24], we assume that $i \in \mathcal{N}_i^{\text{in},t}, \forall t \in \mathbb{N}$, i.e., agent i can always access its own information.

of p_i^0, they can recover an approximation $\hat{f}_i(x) : X \to \mathbb{R}$ for $f_i(x)$. Note that $\hat{f}_i(x)$ is in the form of (5) with its coefficients stored in $\hat{p}_i$. Hence, p_i^0 is the sensitive information of $f_i(x)$ and its privacy needs to be preserved.

In this paper, we aim to design a secure average consensus algorithm for D-CPOA to effectively preserve the privacy of $f_i(x)$, or more specifically, p_i^0. This algorithm will be tailored to the case where agents own local variables of different lengths. To characterize the privacy degree, we use (α, β)-data-privacy, which is a comprehensive measure of the estimation accuracy and disclosure probability [22]. Let $\hat{p}_i$ be the estimation of p_i^0 based on be the available information set $\mathcal{I}$ and the predefined rule. The definition of (α, β)-data-privacy is given as follows.

Definition 1. *A distributed algorithm achieves (α, β)-data-privacy for p_i^0 iff*

$$\Pr\left\{\|\hat{p}_i - p_i^0\|_1 \le \alpha | \mathcal{I}\right\} \le \beta. \tag{6}$$

In the above definition, $\alpha \ge 0$ and $\beta \ge 0$ are parameters that indicate the estimation accuracy and the bound for the disclosure probability of p_i^0, respectively. When α is specified, a smaller β corresponds to a higher degree of privacy preservation. The original definition of (α, β)-data-privacy in [22] considers the estimation of scalar states, and it is extended in this paper to handle vector states. We use the l_1-norm of the error $\hat{p}_i - p_i^0$ to measure the estimation accuracy. This usage contributes to the neat relationship between the estimation accuracy of p_i^0 and that of $f_i(x)$. Detailed discussions are provided in Remark 3.

III. Design of Dependable-CPOA

In this section, we present the design of Dependable-CPOA (D-CPOA). The proposed algorithm consists of three stages, whose details are discussed in the following three subsections.

A. Construction of Local Chebyshev Proxies

In this stage, every agent i computes a polynomial approximation $p_i(x)$ for $f_i(x)$ on $X = [a, b]$, s.t.

$$|f_i(x) - p_i(x)| \le \epsilon_1, \quad \forall x \in [a, b] \tag{7}$$

holds, where $\epsilon_1 > 0$ is a specified tolerance. This goal is achieved by using the adaptive Chebyshev interpolation method [28]. In this method, the degree of the interpolant is systematically increased until a certain stopping criterion is satisfied. The details are as follows. Agent i sets $m_i = 2$ and begins to calculate a Chebyshev interpolant of degree m_i. It evaluates $f_i(x)$ at the $(m_i + 1)$-point grid $S_{m_i} \triangleq \{x_k\}$ by

$$\begin{cases} x_k = \dfrac{b-a}{2} \cos\left(\dfrac{k\pi}{m_i}\right) + \dfrac{a+b}{2}, \\ f_k = f_i(x_k), \end{cases} \tag{8}$$

where $k = 0, 1, \ldots, m_i$. Then, it calculates the coefficients of the interpolant in (5) by

$$c_j = \frac{1}{m_i}\left(f_0 + f_{m_i} \cos(j\pi)\right) + \frac{2}{m_i} \sum_{k=1}^{m_i - 1} f_k \cos\left(\frac{jk\pi}{m_i}\right), \tag{9}$$

where $j = 0, 1, \ldots, m_i$ [28]. At every iteration, the degree m_i is doubled until the stopping criterion

$$\max_{x_k \in \left(S_{2m_i} - S_{m_i}\right)} |f_i(x_k) - p_i(x_k)| \le \epsilon_1 \tag{10}$$

is met, where $p_i(x)$ takes the form of (5) with $\{c_j\}$ being the coefficients. The intersection $X = [a, b]$ of local constraint sets is known by running some numbers of max/min consensus iterations as (2) beforehand.

B. Privacy-Preserving Information Dissemination

After the stage of initialization, each agent owns a local variable $p_i^0 \in \mathbb{R}^{m_i+1}$, which is the vector of coefficients of local polynomial approximation $p_i(x)$. In this stage, the goal is to enable agents to agree on the average $\bar{p} = 1/N \sum_{i=1}^N p_i^0$ of their initial values[2] via a distributed mechanism and, at the same time, the privacy of these initial values is preserved.

We propose a privacy-preserving consensus-based scheme of information dissemination to achieve the aforementioned goal. The backbone of this scheme is the push-sum average consensus protocol. The key ideas of the developed privacy-preserving mechanism are i) adding random noises to p_i^0 to mask the true values, ii) inserting the elements of the perturbed initial states block-by-block to hide them within the iterations, and iii) subtracting the noises separately in several randomly chosen rounds of iterations to guarantee the accuracy of average consensus. The details are as follows.

First, every agent i generates a noise vector $\theta_i \in \Theta^{m_i+1}$ whose elements are independent random variables within the domain Θ, and adds θ_i to its initial state p_i^0 to form a perturbed state $\tilde{p}_i^0$, i.e.,

$$\tilde{p}_i^0 = p_i^0 + \theta_i.$$

Then, agents go on push-sum consensus iterations to exchange and update their local variables x_i^t and y_i^t. The initial value of y_i^t is set as 1 for all $i \in \mathcal{V}$. Nonetheless, instead of directly setting the initial value of x_i^t as $\tilde{p}_i^0$, every agent i will gradually extend x_i^t with the elements of $\tilde{p}_i^0$ in the first K_1 iterations. Let $(d_i^1, \ldots, d_i^{K_1})$ be drawn from the multinomial distribution with parameters $m_i + 1$ and $\left(\frac{1}{K_1}, \ldots, \frac{1}{K_1}\right)$. Then,

$$\sum_{t=1}^{K_1} d_i^t = m_i + 1, \quad d_i^t \in \{0, \ldots, m_i + 1\}, \ \forall t.$$

Hence, $(d_i^1, \ldots, d_i^{K_1})$ can be used to denote the numbers of elements of $\tilde{p}_i^0$ that are inserted to x_i^t at every iteration. Let

$$l_i^0 = 0, \qquad l_i^t = \sum_{k=1}^{t} d_i^k, \ t = 1, \ldots, K_1.$$

At the t-th iteration, the $(l_i^{t-1} + 1)$-th to l_i^t-th elements of $\tilde{p}_i^0$ are inserted into x_i^t to form x_i^{t+}. The rule of insertion is as follows. For all $t = 1, \ldots, K_1$,

$$x_i^{t+}(k) = \begin{cases} x_i^t(k) + \tilde{p}_i^0(k), & \text{for } k = l_i^{t-1} + 1, \ldots, l_i^t, \\ x_i^t(k), & \text{else.} \end{cases} \tag{11}$$

[2] In this expression of the average, those variables of shorter lengths are extended with zeros when necessary to ensure the agreement in dimensions.

Note that if the corresponding $x_i^t(k)$ is null, it is regarded as 0 in (11). Then, agents transmit x_i^{t+} and y_i^t to their out-neighbors and update x_i^{t+1} and y_i^{t+1} by

$$x_i^{t+1}(k) = \sum_{j\in\mathcal{N}_i^{\text{in},t}} a_{ij}^t x_j^{t+}(k), \forall k, \quad y_i^{t+1} = \sum_{j\in\mathcal{N}_i^{\text{in},t}} a_{ij}^t y_j^t. \quad (12)$$

Note that (12) also involves the extension of x_i^t. The length of x_i^{t+1} will be the same as that of the longest $x_j^{t+}, \forall j \in \mathcal{N}_i^{\text{in},t}$. At the end of the K_1-th iteration, all the elements of $\tilde{p}_i^0$ have been gradually inserted, and the length of x_i^t is at least m_i+1. In the following K_2-K_1 iterations, to guarantee the accuracy of the average consensus iterations, every agent will properly subtract the added noises. Let L be a random integer between 1 and K_2-K_1 such that

$$|\zeta_i(k)| > \alpha, \quad \text{where } \zeta_i(k) \triangleq \frac{\theta_i(k)}{L}. \quad (13)$$

Note that L can be drawn from various discrete distributions, e.g., the discrete uniform, binomial and hypergeometric distributions. The choices of such distributions are up to the agents and are unknown to the adversaries. For the k-th element of $x_i^t(\forall k)$, at L randomly selected numbers of iterations, every agent i subtracts a fraction of the added noise $\zeta_i(k)$ from the updated state, i.e.,

$$\begin{aligned} x_i^{t+1}(k) &= \sum_{j\in\mathcal{N}_i^{\text{in},t}} a_{ij}^t x_j^{t+}(k), \\ x_i^{(t+1)+}(k) &= x_i^{t+1}(k) - \zeta_i(k), \quad \forall k, \\ y_i^{t+1} &= \sum_{j\in\mathcal{N}_i^{\text{in},t}} a_{ij}^t y_j^t. \end{aligned} \quad (14)$$

The numbers of these selected iterations form a set $\mathbb{X}_{i,k}, \forall k$. It is assumed that the duration of this period is sufficient for all x_i^t to be extended to the length $m+1$, where

$$m \triangleq \max_{i\in\mathcal{V}} m_i$$

is the maximum degree of all the local approximations. At the rest of the iterations, agents update their local variables by (12), where $x_i^{t+}(k)$ is set as $x_i^t(k), \ \forall k \geq K_2+1$.

To realize distributed stopping when the precision of iterations has met the requirement, we utilize the max/min-consensus-based stopping mechanism in [29] after the K_2-th iteration. Note that the scheme in [29] deals with static digraphs, but it can be easily extended to the settings with time-varying digraphs, given that in this case the max/min consensus protocols can still converge in finite time. The following assumption is required by this mechanism.

Assumption 5. *Every agent i in $\mathcal{G}$ knows an upper bound U on $(N-1)B$, such that U is of the same order as $(N-1)B$.*

Specifically, there are two auxiliary variables, i.e., $r_i^t, s_i^t \in \mathbb{R}^{m+1}$, initialized as $p_i^{K_2} = x_i^{K_2}/y_i^{K_2}$ and updated together with x_i^t and y_i^t by

$$r_i^{t+1}(k) = \max_{j\in\mathcal{N}_i^{\text{in},t}} r_j^t(k), \quad s_i^{t+1}(k) = \min_{j\in\mathcal{N}_i^{\text{in},t}} s_j^t(k), \quad (15)$$

where $k = 1, \ldots, m+1$. These variables are reinitialized as p_i^t every U iterations, so that the recent information of p_i^t is continually disseminated. When the stopping criterion

$$\|r_i^K - s_i^K\|_\infty \leq \delta, \quad \delta = \frac{\epsilon_2}{m+1} \quad (16)$$

is satisfied at the K-th iteration, agents terminate the iterations and set $p_i^K = x_i^K/y_i^K$.

C. Polynomial Optimization by Solving SDPs

In this stage, agents locally optimize the polynomial proxy $p_i^K(x)$ recovered from p_i^K on $X=[a,b]$ to obtain ϵ-optimal solutions of problem (1). This optimization problem can be transformed to a semidefinite program (SDP), and it can be efficiently solved by using the interior-point method [30]. We provide such reformulations based on the Chebyshev coefficients of $p_i^K(x)$.

Note that $p_i^K(x)$ is a polynomial of degree m and takes the form of (5). The elements of $p_i^K = [c_0, \ldots, c_m]^T$ are its coefficients. To simplify the notation, let

$$g_i^K(u) \triangleq p_i^K\Big(\frac{b-a}{2}u + \frac{a+b}{2}\Big) = \sum_{j=0}^m c_j T_j(u), \quad u \in [-1,1].$$

The optimal values of $p_i^K(x)$ on $[a,b]$ and $g_i^K(u)$ on $[-1,1]$ are the same, and the optimal points x_p^* and u_g^* satisfy

$$x_p^* = \frac{b-a}{2}u_g^* + \frac{a+b}{2}. \quad (17)$$

Hence, we solve the following problem

$$\min_u \ g_i^K(u), \quad \text{s.t. } u \in [-1,1], \quad (18)$$

and then use (17) to obtain the optimal value and optimal points of $p_i^K(x)$ on $[a,b]$. To this end, we first transform problem (18) to

$$\max_t \ t, \quad \text{s.t. } g_i^K(x) - t \geq 0, \quad \forall x \in [-1,1]. \quad (19)$$

Then, we introduce new optimization variables $Q, Q' \in \mathbb{S}_+$. When m is odd, problem (19) is transformed to

$$\begin{aligned} \max_{t,Q,Q'} \quad & t \\ \text{s.t.} \quad & c_0 = t + Q_{00} + Q'_{00} + \frac{1}{2}\sum_{u=1}^m (Q_{uu} + Q'_{uu}) \\ & \quad + \frac{1}{4}\sum_{|u-v|=1} (Q_{uv} - Q'_{uv}), \\ & c_j = \frac{1}{2}\sum_{(u,v)\in\mathcal{A}} (Q_{uv} + Q'_{uv}) \\ & \quad + \frac{1}{4}\sum_{(u,v)\in\mathcal{B}} (Q_{uv} - Q'_{uv}), \quad j = 1, \ldots, m, \\ & Q \in \mathbb{S}_+^{\lfloor m/2 \rfloor + 1}, \ Q' \in \mathbb{S}_+^{\lfloor (m-1)/2 \rfloor + 1}, \end{aligned} \quad (20)$$

where the rows and columns of Q and Q' are indexed by $0, 1, \ldots$, and

$$\begin{aligned} \mathcal{A} &= \{(u,v) | u+v = i \vee |u-v| = i\}, \\ \mathcal{B} &= \big\{(u,v) | u+v = i-1 \vee |u-v| = i-1 \\ & \qquad \vee |u+v-1| = i \vee \big||u-v|-1\big| = i\big\}. \end{aligned}$$

When m is even, the transformed problem takes a similar form. We refer readers to our work [17] for more details on the forms and sensibilities of these reformulations[3].

The aforementioned transformed problems are SDPs, and therefore can be efficiently solved via the primal-dual interior-point method [30]. The iterations of this method are terminated when

$$0 \leq f_e^* - p^* \leq \epsilon_3,$$

where f_e^* is the obtained estimate of the optimal value p^* of $p_i^K(x)$ on $X = [a, b]$, and $\epsilon_3 > 0$ is the specified precision. The optimal points of $g_i^K(x)$ are computed from the complementary slackness condition [31]. The optimal points of $p_i^K(x)$ on X can then be calculated by (17).

The full details of the proposed algorithm are summarized as Algorithm 1. We set all the precision used in three stages, i.e., ϵ_1, ϵ_2 and ϵ_3, as $\epsilon/3$, such that their sum equals to ϵ and then the reach of ϵ-optimality is ensured.

IV. Performance Analysis

A. Accuracy

We establish the accuracy of D-CPOA in this subsection. We use ϵ and f^* to denote the specified precision and the optimal value of problem (1), respectively. The following lemma guarantees the accuracy of the consensus iterations within the proposed algorithm.

Lemma 1. *When (16) is satisfied, we have*

$$\max_{i \in \mathcal{V}} \left\| p_i^K - \bar{p} \right\|_\infty \leq \delta, \tag{21}$$

where $\delta = \epsilon_2/(m+1)$.

Proof. The proof is provided in Appendix A. □

Remark 1. *Lemma 1 is in the same form as [17, Theorem 1], but its proof is much more involved. Here we need to prove that with the insertions of perturbed data and separate subtractions of noises, the reach of exact average consensus is still ensured. We also need to verify the effectiveness of the stopping criterion (16) in this case.*

The following theorem demonstrates the accuracy of the proposed algorithm.

Theorem 2. *Suppose that Assumptions 1-5 hold. D-CPOA ensures that every agent obtains ϵ-optimal solutions f_e^* for problem (1), i.e.,*

$$|f_e^* - f^*| \leq \epsilon,$$

where $\epsilon > 0$ is any arbitrarily small specified precision.

Proof. The proof is provided in Appendix B. □

Remark 2. *Theorem 2 takes the same form as [17, Theorem 4]. It implies that even though various challenging requirements concerning privacy and robustness are taken into account, arbitrarily precise globally optimal solutions are still obtained by using the proposed algorithm.*

Algorithm 1 D-CPOA

```
Input: $f_i(x), X_i = [a_i, b_i], U$ and $\epsilon$.
Output: $f_e^*$ for every agent $i \in \mathcal{V}$.
 1: Initialize: $a_i^0 = a_i, b_i^0 = b_i, m_i = 2$.
 2: for each agent $i \in \mathcal{V}$ do
 3:   for $t = 0, \ldots, U-1$ do
 4:     $a_i^{t+1} = \max_{j \in \mathcal{N}_i^{\text{in},t}} a_j^t,\ b_i^{t+1} = \min_{j \in \mathcal{N}_i^{\text{in},t}} b_j^t$.
 5:   end for
 6:   Set $a = a_i^t,\ b = b_i^t$.
      ..................................................
 7:   Calculate $\{x_j\}$ and $\{f_j\}$ by (8).
 8:   Calculate $\{c_k\}$ by (9).
 9:   If (10) is satisfied (where $\epsilon_1 = \epsilon/3$), go to step 10. Otherwise,
      set $m_i \leftarrow 2m_i$ and go to step 7.
      ..................................................
10:   Set $\tilde{p}_i^0 = p_i^0 + \theta_i, x_i^0 = \text{null}, y_i^0 = 1, (d_i^1, \ldots, d_i^{K_1}) \sim$
      $Multi(m_i + 1, 1/K_1(1, \ldots, 1)), l = 1$.
11:   for $t = 0, 1, \ldots$ do
12:     if $t \leq K_1$ then
13:       Extend $x_i^t$ to form $x_i^{t+}$ by (11).
14:       Update $x_i^{t+1}, \forall k$ and $y_i^{t+1}$ by (12).
15:     else if $K_1 + 1 \leq t \leq K_2$ then
16:       for each element $k = 1, \ldots, m_i$ do
17:         if $t \in \mathbb{X}_{i,k}$ then
18:           Update $x_i^{t+1}(k), \forall k$ and $y_i^{t+1}$ by (14).
19:         else
20:           Update $x_i^{t+1}(k), \forall k$ and $y_i^{t+1}$ by (12).
21:         end if
22:       end for
23:     else
24:       if $t = lU$ then
25:         if $\|r_i^t - s_i^t\|_\infty \leq \epsilon_2/(m+1)$ then
26:           $p_i^K = x_i^t / y_i^t$. break
27:         end if
28:         $r_i^t = s_i^t = p_i^t,\ l \leftarrow l + 1$.
29:       end if
30:       Update $x_i^{t+1}(k), \forall k$ and $y_i^{t+1}$ by (3).
31:     end if
32:   end for
      ..................................................
33:   Solve the reformulated problem, e.g, (20), with $\epsilon_3 = \epsilon/3$ and
      return $f_e^*$.
34: end for
```

B. Data-Privacy

In this subsection, we show that the developed algorithm preserves the privacy of p_i^0 and investigate the privacy-preserving property through the notion of data-privacy [22]. We first define the information set $\mathcal{I}_i^t$ used by the adversaries for state estimation. Let

$$\mathcal{I}_i^t = \mathcal{I}_i^{\text{own},t} \bigcup \mathcal{I}_i^{\text{in},t},$$

where

$$\begin{aligned} \mathcal{I}_i^{\text{own},t} &= \bigcup_{s=1}^{t} I_i^{\text{own},s} = \bigcup_{s=1}^{t} \{a_{ii}^s, x_i^{s+}\}, \\ \mathcal{I}_i^{\text{in},t} &= \bigcup_{s \in \mathbb{S}_t} I_i^{\text{in},s} \ = \bigcup_{s \in \mathbb{S}_t} \{a_{ij}^s, x_j^{s+} | j \in \mathcal{N}_i^{\text{in},s}\}. \end{aligned}$$

The set $\mathbb{S}_t$ contains those numbers of iterations $s (s \leq t)$ when the adversaries have obtained the full knowledge of $I_i^{\text{in},s}$. Note that $\mathcal{I}_i^t$ consists of all the available information on the states and weights owned by and transmitted to agent i up to the t-th iteration. Let X be a random variable whose distribution and any other relevant information are unknown. Since X can be

[3] Such SDP reformulations are only dependent on the Chebyshev coefficients of the polynomial $p_i^K(x)$ to be optimized, and are independent of the network topology. Hence, the analysis of the reformulations in [17], which considers static undirected networks, also applies to this paper.

any arbitrary value in its domain, it is reasonable to assume that the probability that an accurate enough estimation $\hat{X}$ of X can be obtained is rather small [27], i.e.,

$$\Pr\left\{|\hat{X}-X|\le\alpha\right\}\le\gamma, \tag{22}$$

where $\alpha\ge 0$ and $\gamma\ge 0$ are small given constants. The bound γ for the disclosure probability satisfies

$$\gamma \ll p\max_{\nu\in\Theta}\int_{\nu-\alpha}^{\nu+\alpha} f_{\theta_i(k)}(y)\mathrm{d}y, \quad \forall k,$$

where $f_{\theta_i(k)}(y)$ is the probability density function (PDF) of the added noise $\theta_i(k)$.

Recall that we aim to preserve the privacy of $p_i^0\in\mathbb{R}^{m_i+1}$. Let α and α_k be the estimation accuracy of p_i^0 and each of the element $p_i^0(k)$, respectively, s.t.,

$$\sum_{k=1}^{m_i+1}\alpha_k=\alpha, \quad \alpha_k\ge 0,\ \forall k=1,\ldots,m_i+1. \tag{23}$$

The following theorem characterizes the effects of privacy preservation of the developed algorithm.

Theorem 3. *If Assumptions 3 and 4 hold, D-CPOA achieves (α,β)-data-privacy for p_i^0, where*

$$\beta=\prod_{k=1}^{m_i+1}\Big[\big(1-p^{K_2-K_1}\big)h_i(\alpha_k)+p^{K_2-K_1}\Big], \tag{24}$$

and

$$h_i(\alpha_k)=p\max_{\nu\in\Theta}\int_{\nu-\alpha_k}^{\nu+\alpha_k} f_{\theta_i(k)}(y)\mathrm{d}y+\gamma,$$

and $\{\alpha_k\}$ satisfies (23).

Proof. The proof is provided in Appendix C. □

Theorem 3 states that D-CPOA preserves the privacy of p_i^0. The effects of privacy preservation are evaluated through (α,β)-data-privacy. The interpretation of β in (24) is as follows. Note that β is the product of a set of bounds β_k for disclosure probabilities corresponding to the elements $p_i^0(k),\forall k=1,\ldots,m_i+1$ (see (35) and (36)). The bounds β_k are derived via the law of total probability. If the event that the adversaries know $I_i^{\text{in},t}$ for any time t between K_1+1 and K_2 happens (the probability of which is not more than $p^{K_2-K_1}$), then the added noises $\theta_i(k)$ and states $p_i^0(k)$ can be perfectly inferred. Otherwise, the disclosure probability will not exceed $h_i(\alpha_k)$. The bounds $h_i(\alpha_k)$ are derived likewise based on whether the adversaries know $I_i^{\text{in},s-1}$, where s is the time when agent i inserts its perturbed state $\tilde{p}_i^0(k)$. If the adversaries know this information, then the maximum disclosure probability is

$$\max_{\nu\in\Theta}\int_{\nu-\alpha_k}^{\nu+\alpha_k} f_{\theta_i(k)}(y)\mathrm{d}y,$$

which equals to the probability that the optimal distributed estimation falls into $[p_i^0(k)-\alpha_k, p_i^0(k)+\alpha_k]$ [22]. Otherwise, the disclosure probability is rather small since the adversaries own little relevant information of p_i^0.

From (24), we know that for those p_i^0 of longer lengths (i.e., with larger m_i), β will generally be smaller, which implies a higher degree of privacy preservation. In addition, β increases with α_k but decreases with K_2-K_1. These relationships support the intuitions that less accurate estimations can be acquired with higher probabilities, and more room for randomness leads to lower probabilities of privacy disclosure.

Remark 3. *In this paper, we investigate the privacy-preserving property of D-CPOA by studying its degree of data-privacy for p_i^0. The reasons are twofold. First, this degree directly reflects the effectiveness of the incorporated privacy-preserving mechanism, since p_i^0 is exactly the initial value calling for protections in the iterations. Second, this degree is closely related to the effects of privacy preservation of $f_i(x)$. In (6), if $\|\hat{p}_i-p_i^0\|_1\le\alpha$, i.e., a fairly precise estimation $\hat{p}_i$ of p_i^0 is obtained, then $\forall x\in X=[a,b]$, we have*

$$\begin{aligned}|\hat{f}_i(x)-p_i(x)|&=\left|\sum_{k=0}^{m}(\hat{p}_i(k)-p_i^0(k))T_k\Big(\frac{2x-(a+b)}{b-a}\Big)\right|\\&\le\sum_{k=0}^{m}|\hat{p}_i(k)-p_i^0(k)|\cdot 1=\|\hat{p}_i-p_i^0\|_1\le\alpha.\end{aligned} \tag{25}$$

By referring to (7), it follows that

$$|\hat{f}_i(x)-f_i(x)|\le\alpha+\epsilon_1,$$

i.e., an accurate enough estimation $\hat{f}_i(x)$ of $f_i(x)$ is acquired. Nevertheless, to derive the requirement of closeness between $\hat{p}_i$ and p_i^0 from that between $\hat{f}_i(x)$ and $f_i(x)$(or $p_i(x)$) is very difficult due to the coupling of terms in (25). Hence, the characterization of the degree of data-privacy for p_i^0 is the main focus of this paper.

C. Discussions on Dependability

In this section, we discuss the dependability of the proposed algorithm, considering various requirements including privacy preservation and robustness to network imperfections. We summarize the comparisons of the performance of D-CPOA and other typical algorithms in Table I. The details are given as follows.

- *Privacy Guarantee.* We have shown in Theorem 3 that the consensus-based iterations of D-CPOA preserve the privacy of sensitive p_i^0 and analyzed the effects of preservation through the notion of (α,β)-data-privacy. Next, we study such effects via *differential privacy*, which provides a strong privacy guarantee when in face of adversaries owning arbitrarily much auxiliary information [4], [19], [33]. Let

$$P=\{p_i^0|\forall i\in\mathcal{V}\}$$

be the dataset of all the initial states and

$$\mathcal{M}(P)=\{x_i^+(t)|\forall t\in\mathbb{N}, i\in\mathcal{V}\},$$

i.e., the set of transmitted states of consensus protocols, be the randomized query output. By referring to [19], [33], in our setting, a privacy-preserving consensus protocol is ϵ-*differentially private* if

$$\Pr\left\{\mathcal{M}(P)\in\mathcal{O}\right\}\le e^{\epsilon}\Pr\left\{M(P')\in\mathcal{O}\right\}$$

holds for any $\mathcal{O}\subseteq\text{range}(\mathcal{M})$ and σ-adjacent P,P' satisfying

$$\left\|p_i^0-(p_i^0)'\right\|_1\le\begin{cases}\sigma, & \text{if } i=i_0,\\ 0, & \text{if } i\ne i_0\end{cases}$$

TABLE I
COMPARISONS OF D-CPOA AND OTHER TYPICAL DISTRIBUTED OPTIMIZATION ALGORITHMS

Algorithms	**Nonconvex Objectives**	**Networks**		**Privacy Guarantee**	**Asynchrony**	**Exact Convergence**	**Complexities**
		Time-varying	Digraph				
Push-DIGing [11]		✓	✓			✓	scvx[1]: linear
G-Push-Pull [13]			✓		✓	mean-square	scvx: linear
SONATA [32]	✓	✓	✓			✓	scvx: linear ncvx[2]: $\mathcal{O}\big(\frac{1}{\epsilon}\big)$[3]
ASY-SONATA [16]	✓		✓		✓	✓	scvx: linear ncvx: $\mathcal{O}\big(\frac{1}{\epsilon}\big)$
Algorithm in [4]		Cloud-based		DP[4]		trade-off[5]	
Algorithm in [9]		Cloud-based		DP		—"—[6]	
Algorithm in [10][7]		✓	✓	DP	✓	—"—	
D-CPOA	✓	✓	✓	(α, β)- data-privacy (Theorem 3)	✓[8]	✓	0^{th}-ord. oracle: $\mathcal{O}(m)$ Commn.: $\mathcal{O}\big(\log\frac{m}{\epsilon}\big)$ PD itr.: $\mathcal{O}\big(\sqrt{m}\log\frac{1}{\epsilon}\big)$[9]

[1] "scvx" refers to "strongly-convex" objective functions. [2] "ncvx" refers to "nonconvex" objective functions.
[3] The convergence time is $\mathcal{O}\big(\frac{1}{\epsilon}\big)$, implying that both the complexities of evaluations of local gradients (i.e., queries of the first-order oracle) and those of inter-agent communication are $\mathcal{O}\big(\frac{1}{\epsilon}\big)$. [4] "DP" stands for "differential privacy". [5] There is a trade-off between accuracy and privacy.
[6] This symbol stands for "same as above". [7] In [10], the authors propose a general strategy of function perturbation to achieve differential privacy in distributed optimization. This strategy can be combined with any distributed convex constrained optimization algorithms to take effect. Hence, we place "✓" to some blocks in this row to imply feasible possibilities. [8] Detailed discussions are provided in Sec. IV-C. [9] See Theorem 4 for details.

for all $i \in \mathcal{V}$, where i_0 is some element in $\mathcal{V}$. Note that we have used correlated noises (see (13)) to pursue the proximity of p_i^K to the exact average $\bar{p}$ (see Lemma1), thus ensuring the accuracy of the obtained solutions (see Theorem3). Based on the impossibility result of simultaneously achieving exact average consensus and differential privacy [19], [33], we conclude that the current algorithm is not ϵ-differentially private. If we want to preserve differential privacy at the cost of losing some accuracy of the obtained solutions, we can add uncorrelated noises that satisfy the condition in [33, Theorem 4.3] to the transmitted states at every iteration. In this case, the almost sure limit of differentially private consensus iterations is an unbiased estimate of the exact average. The random difference between the limit and the average will lead to an additional random error in the returned solutions of the proposed algorithm.

Remark 4. *Existing privacy-preserving distributed optimization algorithms (e.g., [4], [9], [10]) mainly use uncorrelated noises to perturb the exchanged messages and are differentially private. The notion of differential privacy provides strong privacy guarantees. Also, its nice property of sequential composability facilitates the analysis of privacy when confronted with complex and nonlinear iterations involving gradients. Nonetheless, there always exists a trade-off between privacy and accuracy [10], which calls for a careful selection of related parameters to obtain a rather small bound on the sub-optimality. In contrast, due to the simple and linear consensus-based iterations of the proposed algorithm, we can either use correlated noises to readily achieve the effective preservation of privacy and ensure the accuracy of the obtained solutions, or use uncorrelated noises to pursue the strong guarantee offered by differential privacy.*

- *Asynchrony.* We discuss the asynchronous extension of the proposed algorithm. Compared to synchronous models, asynchronous paradigms are more desirable in applications for its increased efficiency in handling uncoordinated computations and imperfect communication, e.g., transmission delays and packet drops. The design of consensus-based information dissemination presented in Algorithm 1 is synchronous. Its extension to cope with asynchrony is readily available and can benefit from the extensive research on asynchronous consensus protocols, including those allowing for random activations (e.g., gossip algorithms [34]), delays [35], packet drops [36] and all these issues [16]. In these protocols, the basic idea of proving convergence is to first transform asynchronous models to synchronous counterparts over augmented graphs, where virtual nodes and edges are added to facilitate the analysis, and then establish the convergence of synchronous models. All these asynchronous protocols converge deterministically to the average of initial values. If they are incorporated into the proposed algorithm, by Lemma 1 and Theorem 2, the accuracy of the obtained solutions can still be guaranteed. In addition, since the iterations of the developed algorithm are consensus-based and do not involve gradients, there is no need to select varying step-sizes in different circumstances of asynchrony.

D. Complexity

The following theorem describes the complexities of the developed algorithm.

Theorem 4. *D-CPOA ensures that every agent obtains ϵ-optimal solutions for problem (1) with $\mathcal{O}(m)$ evaluations of local objective functions, $\mathcal{O}\big(\log\frac{m}{\epsilon}\big)$ rounds of inter-agent communication and $\mathcal{O}\big(\sqrt{m}\log\frac{1}{\epsilon}\big)$ iterations of primal-dual*

TABLE II
COMPARISONS OF COMPLEXITIES

Alg.	0^{th}-order Oracles	Communication	PD Iterations
CPCA	$\mathcal{O}(m)$	$\mathcal{O}\Big(\log\dfrac{m}{\epsilon}\Big)$	$\mathcal{O}\Big(\sqrt{m}\log\dfrac{1}{\epsilon}\Big)$
D-CPOA	$\mathcal{O}(m)$	$\mathcal{O}\Big(\log\dfrac{m}{\epsilon}\Big)$[1]	$\mathcal{O}\Big(\sqrt{m}\log\dfrac{1}{\epsilon}\Big)$

[1] Compared with CPCA, D-CPOA generally requires more inter-agent communication to reach certain precision. This increase results from potential network imperfections and the extra steps of insertions and subtractions, which may slow down the convergence rates. Nonetheless, the communication complexities of both algorithms are the same (see proof of Theorem 4).

interior-point methods[4].

Proof. Note that the evaluations of local objective functions (i.e., queries of the zeroth-order oracle) are only performed in the stage of initialization, and the primal-dual interior-point method [30] is used to solve problem (20) in the stage of polynomial optimization. By referring to the proof of [17, Theorem 6], we know that for every agent, the orders of evaluations of local objective functions and primal-dual iterations are $\mathcal{O}(m)$ and $\mathcal{O}\big(\sqrt{m}\log\frac{1}{\epsilon}\big)$, respectively, where m is the maximum degree of local approximations.

In the stage of information dissemination, the insertions of block-data and subtractions of noises are completed in finite time, i.e., within K_2 iterations. Since the consensus-based protocol converges geometrically, the order of the total number of iterations (i.e., inter-agent communication) is

$$K_2 + \mathcal{O}\Big(\log\frac{1}{\delta}\Big) = \mathcal{O}\Big(\log\frac{1}{\delta}\Big) = \mathcal{O}\Big(\log\frac{m}{\epsilon}\Big),$$

where the required precision δ is given by (16). □

The comparisons of the complexities of D-CPOA and those of CPCA [17] are shown in Table II. We observe that the complexities of these two algorithms are the same. The reasons are as follows. The major difference between the two algorithms lies in the stage of consensus-based information dissemination. In this stage, D-CPOA fulfills privacy preservation by utilizing the randomness of insertions of block-data and subtractions of added noises. These actions are completed in finite time, and thus they only change the values but not the orders of the needed numbers of iterations (i.e., inter-agent communication). Hence, we conclude that the dependability of the proposed algorithm brings no extra costs in terms of complexities.

E. Discussions on Multivariate Extensions

In this paper, we mainly consider problems with univariate objective functions to highlight the various advantages brought by the idea of using polynomial approximation, e.g., achieving efficient optimization of nonconvex problems and readily allowing for enhancement to be dependable when diverse practical needs exist. We now briefly discuss the possibility of multivariate extensions of the proposed idea. The differences will mainly lie in the stage of initialization and that of optimization of approximations. Specifically, let $L_2(X)$ be the set of square-integrable functions over $X \subset \mathbb{R}^n$ and $f_i(x) \in L_2(X)$ be a general local objective function. Then, there exists an orthonormal basis $\{h_k(x)\}_{k\in\mathbb{N}_+}$ (e.g., Taylor polynomials) and an arbitrarily precise approximation

$$\hat{f}_i(x) = \sum_{k=1}^{m} c_k h_k(x)$$

for $f_i(x)$, where $\{c_k\}_{k=1}^m$ is the set of coefficients. Afterward, agents can exchange and update their local variables storing these coefficients (as in Sec. III-B) and acquire an approximation for the global objective function. Finally, they can locally optimize this approximation via the tools for polynomial optimization or for finding stationary points of general nonconvex functions, thus obtaining desired solutions. Nevertheless, the aforementioned idea of extensions calls for further investigation and more careful analysis and is still among our ongoing work.

V. NUMERICAL EVALUATIONS

In this section, we perform numerical experiments to illustrate the performance of D-CPOA. We consider a network with $N = 20$ agents. At each time t, besides itself, every agent i has two out-neighbors. One belongs to a fixed cycle, and the other is chosen uniformly at random. Hence, $\{\mathcal{G}^t\}$ is 1-strongly-connected. We assume that all the local constraint sets are the same interval $X = [-1, 1]$ and the local objective function $f_i(x)$ of agent i is

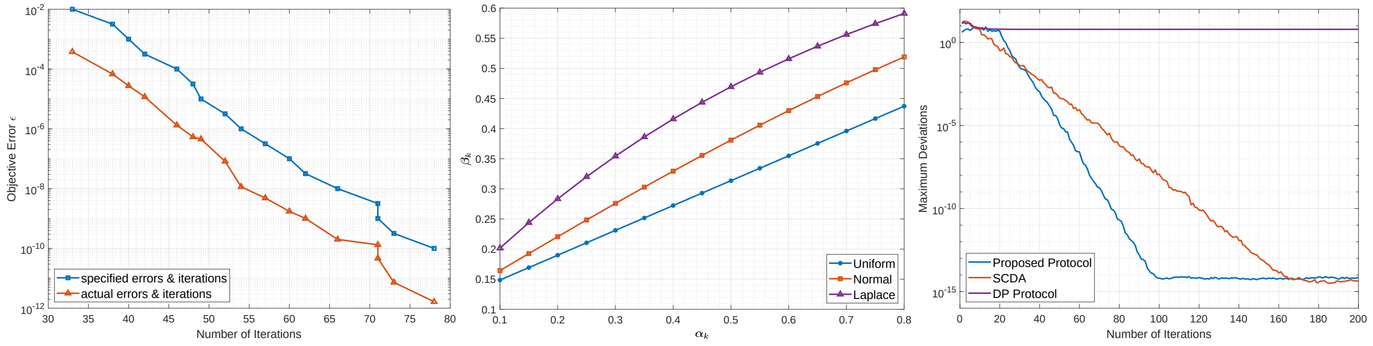
$$f_i(x) = \frac{a_i}{1+e^{-x}} + b_i \log(1+x^2),$$

where $a_i \sim \mathcal{N}(10, 2)$ and $b_i \sim \mathcal{N}(5, 1)$ are normally distributed. It follows that $f_i(x)$ is nonconvex and Lipschitz continuous on X. Chebfun toolbox [26] is used to construct Chebyshev polynomial approximations $p_i(x)$ corresponding to all the local objective functions $f_i(x)$.

The convergence of the proposed algorithm is shown in Fig. 1(a). In the experiment, we set $K_1 = 10, K_2 = 20$. We generate i.i.d. random noises $\theta_i(k)$ from the uniform distribution $\mathcal{U}(-1, 1)$ and randomly select L from the discrete uniform distribution $\mathcal{U}\{1, K_2 - K_1\}$ to satisfy (13). In Fig. 1(a), the square markers on the blue line indicate how many numbers of iterations t of information dissemination have been performed, when certain precisions ϵ are specified. The triangle markers on the orange line represent what the actual values of objective errors $|f_e^* - f^*|$ are, when those numbers of iterations are completed. We observe that the relationship between $\log\epsilon$ and t is roughly linear. This phenomenon results from the property of linear convergence of the consensus-based information dissemination in the developed algorithm.

The effects of privacy preservation are presented in Fig. 1(b). This figure demonstrates the relationships between estimation accuracy α_k and bound β_k for the disclosure probability for a single element $p_i^0(k)$ when different types of noises $\theta_i(k)$ are used. These relationships are explicitly characterized by (35) in Appendix C. In the experiment, we set $K_1 = 10, K_2 = 20, p = 0.8$ and $\gamma = 10^{-5}$. We consider

[4] The dependence of m on ϵ and the smoothness of local objective functions is discussed in [17, Lemma 7].



(a) (b) (c)

Fig. 1. Performance of D-CPOA. (a) Convergence. (b) (α, β)-data-privacy. (c) Convergence of the proposed protocol in Sec. III-B and other privacy-preserving consensus protocols.

three types of noises that satisfy uniform, normal and Laplace distributions. We assume that the mean and variance of these noises are 0 and 1, respectively. We observe that β_k increases with α_k, which confirms the intuition that a less accurate estimate can be obtained with a higher probability. We also notice that uniformly distributed noises yield the smallest β_k and thus the most effective preservation of $p_i^0(k)$. This observation supports the conclusion in [22]. Note that the bound β for the disclosure probability of p_i^0 is the product of all β_k for $k = 1, \ldots, m_i + 1$ (see (36) in Appendix C). The degrees m_i of local approximations constructed in this experiment roughly vary from 20 to 30 when the specified precision $\epsilon = 10^{-10}$. Hence, in this case, β is an extremely small number given α_k and β_k in the figure.

Specifically, we study the convergence rates of the consensus-based iterations incorporated with the proposed privacy-preserving mechanism. The initial states of agents are set as the vectors of coefficients of local approximations. The rest of the settings are the same as those in the study of the convergence of D-CPOA. We also implement SCDA in [20] and a differentially private consensus protocol [19], [33] for comparison, where uniformly distributed and Laplace distributed noises are added to every element of local variables, respectively. In all these three protocols, the initially added noises are of zero mean and variance $1/3$. In the last two protocols, the variances of the added noises decay at a rate of 0.64. The relationships between maximum deviations $\max_{i\in\mathcal{V}} \|p_i^t - \bar{p}\|_1$ and numbers of iterations t are shown in Fig. 1(c). It is observed that our protocol converges faster than SCDA to the exact average. The main reason is that we do not continuously add noises to local variables all along the iterations. Hence, the possible negative effects of noises on the convergence rates are mitigated. Also, the deviation of the differentially private consensus protocol does not converge to 0. This phenomenon reflects the fundamental trade-off between privacy and accuracy in this class of protocols.

VI. Related Work

There have been extensive researches on designing efficient distributed optimization algorithms, e.g., primal methods [5], [6], [14], [37] and dual-based methods [7], [8], [38]. The core idea of the primal methods is to combine consensus with gradient-based optimization algorithms, thus achieving consensual iterative convergence in the primal domain. Thanks to the development of gradient tracking [6], [14], [37], [39], which enables local agents to approximately track the gradients of the global objective function, the convergence rates of these distributed algorithms can nearly match that of the optimal centralized gradient-based algorithm [40]. The basic intuition of the dual-based methods is to express the consensus requirement as equality constraints, and then solve the dual problems of the equivalent reformulations or carry on primal-dual updates. These carefully constructed dual problems are decoupled, thus easily allowing for the distributed implementations of certain linearly convergent centralized optimization algorithms, e.g., ADMM [2]. For convex problems, distributed algorithms guarantee convergence to globally optimal points; for nonconvex problems, the convergence to stationary or locally optimal points is ensured [32], [39], [41]–[43].

The aforementioned work mainly centers on bridging the gap in terms of convergence behaviors between distributed and centralized optimization algorithms. To effectively deploy these distributed algorithms into applications, some specific issues need to be addressed. These issues include but are not limited to privacy preservation, time-varying and directed communication, asynchronous computations due to lack of coordination, transmission delays or packet drops.

Specifically, the privacy concern of distributed algorithms has received growing attention. Conventional approaches are based on the premise that exact local data is exchanged between agents. Nevertheless, if there exist adversaries that intentionally gather certain data necessary for estimation, the sensitive information of objective functions, constraints and local states can be disclosed [4]. To tackle this problem, a number of privacy-preserving consensus and distributed optimization algorithms have been proposed. One typical approach based on the idea of message perturbation is to add random noises to the data transmitted within iterations. The perturbation of the critical data (e.g., states [18]–[20], [27], gradients [4], [9] and functions [10]) limits its utility for yielding sensible estimations. Some work considers the use of uncorrelated Laplacian or Gaussian noises and develops various differentially private consensus [19], [44] and distributed optimization algorithms [4], [9], [10]. The differentially private mechanism equips these algorithms with strong privacy guarantees even against those adversaries owning arbitrarily much side information. Nonetheless, it also brings about the trade-off between privacy and accuracy [10], [19]. Other work thus turns to correlated noises and shows that the exact average consensus is reached [18], [20], [27]. The effects of privacy preservation can be characterized by using the notion of data-privacy [22]. Another typical approach is to apply cryptographic techniques, e.g., homomorphic encryption. Related algorithms can be found in [45]–[47]. These cryptography-based methods are suitable if the requirements of trusted agents or shared keys/secrets are satisfied, and the extra communication and computation burdens induced by encryption and decryption are acceptable.

In addition to the privacy concern, the robustness issues of distributed optimization have also been widely investigated. Time-varying and directed communication inhibits the efficient construction of doubly stochastic weight matrices, which are crucial for achieving convergence over undirected graphs. To overcome this challenge, push-sum-based algorithms [11], [32], [48], [49] and push-pull-based algorithms [12], [13] are developed. The former combine the push-sum consensus protocol [23] with gradient-based methods and only require column stochastic weight matrices. The latter use one row stochastic and one column stochastic weight matrix to mix estimates of optimal solutions and trackers of average gradients, respectively. Algorithms that purely handle random transmission delays can be found in [3], [50], where the basic idea is to locally fuse the delayed information as soon as it arrives. To achieve asynchronous computations, gossip-type algorithms [13], [14] and those further allowing delays and packet drops [15], [16] have been developed.

Different from the aforementioned work, the proposed algorithm is based on the idea of using polynomial approximation and is equipped with effective mechanisms to meet diverse practical requirements concerning privacy and robustness. We show that the efficient distributed optimization of general nonconvex problems is achieved, and in the meantime the common issues of privacy-accuracy trade-off and step-size selections are avoided.

VII. Conclusion

In this paper, we proposed D-CPOA to solve a class of constrained distributed nonconvex optimization problems, considering the needs of privacy preservation and robustness to various network imperfections. We achieved exact convergence and effective preservation of the privacy of local objective functions by incorporating a new privacy-preserving mechanism for consensus-based iterations. The developed mechanism utilized the randomness in block-by-block insertions of perturbed data and separate subtractions of added noise, and its privacy degree was explicitly characterized through (α, β)-data-privacy. We ensured the robustness of the proposed algorithm by using the push-sum average consensus protocol as a basis for iterations, and discussed its extensions to help to maintain the performance when diverse imperfections in network communication exist. We proved that the major benefits brought by the idea of using polynomial approximation were preserved, and the aforementioned demanding requirements were satisfied at the same time.

Appendix

A. Proof of Lemma 1

Proof. The proof consists of two steps. First, we prove that the limit value of $p_i^t \triangleq x_i^t / y_i^t (t \in \mathbb{N})$ is indeed $\bar{p}$, i.e.,

$$\lim_{t \to \infty} p_i^t = \bar{p}. \tag{26}$$

Then, we demonstrate that the meet of the stopping criterion (16) is a sufficient condition for (21).

- *Step 1: Proof of the Limit Value*

We consider the k-th element of the involved local variables, $\forall k = 1, \ldots, m$. Let

$$\begin{aligned} x^t &\triangleq [x_1^t(k), \ldots, x_N^t(k)]^T, \quad \theta \triangleq [\theta_1(k), \ldots, \theta_N(k)]^T, \\ p^0 &\triangleq [p_1^0(k), \ldots, p_N^0(k)]^T, \quad y^t \triangleq [y_1^t, \ldots, y_N^t]^T. \end{aligned}$$

Note that if the k-th elements of some x_j^t, θ_j and $p_j^0 (j \in \mathcal{V})$ are null, they are regarded as 0 in the expressions. We investigate the effects of insertions (11) and subtractions (14) on the accuracy of the consensus-based updates in Algorithm 1 as follows.

We first consider the effect of insertions that happened in the first K_1 iterations. Let t_k be the number of the iteration where agent i inserts the perturbed state $\tilde{p}_i^0(k)$. Since A^{t_k} is column stochastic, from (11) and (12), we have

$$1^T x^{t_k+1} = 1^T A^{t_k} x^{t_k+} = 1^T x^{t_k+} = 1^T x^{t_k} + \tilde{p}_i^0(k),$$

which means that the sum of the elements of x^t increases by $\tilde{p}_i^0(k)$. At the end of the K_1-th iteration, all the agents have inserted their perturbed initial states. Hence,

$$1^T x^{K_1} = 1^T x^0 + \sum_{i \in \mathcal{V}} \tilde{p}_i^0(k) = \sum_{i \in \mathcal{V}} \tilde{p}_i^0(k) = 1^T (p^0 + \theta).$$

Then, we focus on the effect of subtractions happened between time $K_1 + 1$ and time K_2. Suppose that the smallest element in $\mathbb{X}_{i,k}$ is t_1, i.e., agent i performs its first action of subtractions at the t_1-th iteration. From the column stochasticity of $A^t (t \in \mathbb{N})$ and (3), it is not difficult to obtain that

$$1^T x^{t_1} = 1^T A^{t_1 - 1} x^{t_1 - 1} = 1^T x^{t_1 - 1} = \ldots = 1^T x^{K_1}.$$

At the t_1-th iteration, we have

$$1^T x^{t_1 + 1} = 1^T x^{t_1} - \delta_i(k) = 1^T x^{K_1} - \delta_i(k),$$

which implies that the sum of the elements of x^t decreases by $\delta_i(k) = \theta_i(k)/L$. At the end of the K_2-th iteration, every agent has completed its L rounds of actions of subtracting the noises. It follows that

$$1^T x^{K_2} = 1^T x^{K_1} - 1^T \theta = 1^T p^0.$$

Since y_i^t is constantly updated by (3), we have

$$1^T y^{K_2} = 1^T A^{K_2 - 1} y^{K_2 - 1} = 1^T y^{K_2 - 1} = \ldots = 1^T y^0.$$

Later on, agents continue to update x_i^t and y_i^t by (3). Based on the convergence of the push-sum consensus protocol, we conclude that the exact average can still be achieved, i.e.,

$$\begin{aligned}\lim_{t\to\infty} p_i^t &= \lim_{t\to\infty} \frac{x_i^t}{y_i^t} = \frac{1^T x^{K_2}}{1^T y^{K_2}} = \frac{1^T p^0}{1^T y^0} \\ &= \frac{1}{N}\sum_{j=1}^{N} p_j^0(k) = \bar{p}(k).\end{aligned}$$

Note that this result holds for any $k = 1, \ldots, m$. Therefore, the limit value of p_i^t is $\bar{p}$, i.e., (26) holds.

- *Step 2: Proof of the Sufficiency*

Next, we verify the effectiveness of the stopping criterion (16). Note that $p_i^t = x_i^t / y_i^t$, $\forall t \in \mathbb{N}$. The push-sum-consensus-based update of x_i^t in (3) can be transformed to

$$p_i^{t+1} = \sum_{j=1}^{N} w_{ij}^t p_j^t, \quad \text{where } w_{ij}^t = \frac{a_{ij}^t y_j^t}{y_i^{t+1}}.$$

It follows from (3) and (4) that $W^t \triangleq (w_{ij}^t)_{N\times N}$ is row stochastic, i.e.,

$$\sum_{j=1}^{N} w_{ij}^t = 1, \quad 0 \le w_{ij} \le 1, \quad \forall i, j = 1, \ldots, N, \ \forall t.$$

Hence, we have

$$\begin{aligned}p_i^{t+1}(k) &= \sum_{j=1}^{N} w_{ij}^t p_j^t(k) \le \sum_{j=1}^{N} w_{ij}^t \max_{j\in\mathcal{V}} p_j^t(k) \\ &= \max_{j\in\mathcal{V}} p_j^t(k), \quad \forall k = 1, \ldots, m+1, \ \forall i \in \mathcal{V}.\end{aligned}$$

Let $M^t(k) \triangleq \max_{i\in\mathcal{V}} p_i^t(k)$, $m^t(k) \triangleq \min_{i\in\mathcal{V}} p_i^t(k)$. It follows that

$$M^{t+1}(k) \le M^t(k), \quad m^{t+1}(k) \ge m^t(k).$$

It has been proven that $\lim_{t\to\infty} p_i^t(k) = \bar{p}(k), \ \forall i \in \mathcal{V}$. Hence,

$$\lim_{t\to\infty} M^t(k) = \bar{p}(k), \quad \lim_{t\to\infty} m^t(k) = \bar{p}(k).$$

Since the sequences of $\big(M^t(k)\big)_{t\in\mathbb{N}}$ and $\big(m^t(k)\big)_{t\in\mathbb{N}}$ are non-increasing and non-decreasing, respectively, we have

$$m^t(k) \le \bar{p}(k) \le M^t(k), \quad \forall t \in \mathbb{N}.$$

Note that the max/min consensus protocols converge within U iterations. When agents terminate the iterations at time K, we have

$$r_i^K(k) - s_i^K(k) = M^{K'}(k) - m^{K'}(k),$$

where $K' \triangleq K - U$. The meet of the stopping criterion (16) implies that

$$\begin{aligned}\big|p_i^K(k) - \bar{p}(k)\big| &\le M^K(k) - m^K(k) \\ &\le r_i^K(k) - s_i^K(k) \le \delta, \ \forall i, k.\end{aligned}$$

□

B. Proof of Theorem 2

Proof. The proof is rather similar to that of Theorem 4 in [17]. We provide a sketch of the main steps here. The key idea is to prove the closeness between $p_i^K(x)$ and $f(x)$ on the entire $X = [a, b]$. Then, their optimal values are also close enough (see [17, Lemma 3]). Note that $p_i^K(x)$ and $\bar{p}(x)$ are in the forms of (5) with the coefficients stored in p_i^K and $\bar{p}$, respectively. It follows from (21) that

$$\begin{aligned}\big|p_i^K(x) - \bar{p}(x)\big| &= \left|\sum_{j=0}^{m} (c_j - \bar{c}_j) T_j\Big(\frac{2x - (a+b)}{b-a}\Big)\right| \\ &\le \sum_{j=0}^{m} |c_j - \bar{c}_j| \cdot 1 \le \sum_{j=0}^{m} \|p_i^K - \bar{p}\|_\infty \\ &\le \delta(m+1) = \epsilon_2, \qquad \forall x \in [a, b],\end{aligned}$$

where the first inequality is based on $|T_j(u)| \le 1, \forall u \in [-1, 1]$. Note that $\bar{p}$ is the average of all p_i^0. Hence, $\bar{p}(x)$ is also the average of all $p_i(x)$. Based on (7), we have

$$\begin{aligned}|\bar{p}(x) - f(x)| &= \left|\frac{1}{N}\sum_{i=1}^{N} \big(p_i(x) - f_i(x)\big)\right| \\ &\le \frac{1}{N}\sum_{i=1}^{N} |p_i(x) - f_i(x)| \le \frac{1}{N} N\epsilon_1 = \epsilon_1, \quad \forall x \in [a, b].\end{aligned}$$

Given that $\epsilon_1 = \epsilon_2 = \epsilon/3$, we have

$$\begin{aligned}\big|p_i^K(x) - f(x)\big| &\le \big|p_i^K(x) - \bar{p}(x)\big| + |\bar{p}(x) - f(x)| \\ &\le \epsilon_1 + \epsilon_2 = \frac{2}{3}\epsilon, \qquad \forall x \in [a, b].\end{aligned}$$

Let p^* be the optimal value of $p_i^K(x)$ on $X = [a, b]$. It follows from [17, Lemma 3] that

$$|p^* - f^*| \le \frac{2}{3}\epsilon.$$

Note that $p^* \le f_e^* \le p^* + \epsilon_3 = p^* + \frac{\epsilon}{3}$. Hence,

$$f^* - \frac{2}{3}\epsilon \le p^* \le f_e^* \le p^* + \frac{\epsilon}{3} \le f^* + \epsilon,$$

which leads to $|f_e^* - f^*| \le \epsilon$. □

C. Proof of Theorem 3

Proof. We consider the estimation of $p_i^0(k)$, $\forall k = 1, \ldots, m_i + 1$. Suppose that at the t_k-th iteration, agent i inserts the perturbed state $\tilde{p}_i^0(k)$ by (11). Note that the estimation $\hat{p}_i(k)$ of $p_i^0(k)$ can be calculated at three types of time, i.e., before t_k, at t_k and after t_k. We discuss each of these scenarios in detail as follows.

At time $t < t_k$, $\tilde{p}_i^0(k)$ has not been inserted yet. What the adversaries have collected are either null values or combinations of the perturbed states of agent i's neighbors. Since there is not any available information on $p_i^0(k)$ that serves as a basis for the estimation, by (22), we have

$$\Pr\left\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^t\right\} \le \gamma.$$

At time $t = t_k$, $\tilde{p}_i^0(k)$ is inserted. By Assumption 4, the probability that the adversaries acquire the full knowledge of

I_i^{in,t_k-1} is not more than p. If this is the case, based on (11) and (12), they can easily calculate $\tilde{p}_i^0(k)$ by

$$\tilde{p}_i^0(k) = x_i^{t_k+}(k) - \sum_{j\in\mathcal{N}_i^{\text{in},t_k-1}} a_{ij}^{t_k-1} x_j^{(t_k-1)+}(k). \tag{27}$$

Note that

$$\tilde{p}_i^0(k) = p_i^0(k) + \theta_i(k).$$

Hence, after an estimation $\hat{\theta}_i(k)$ of $\theta_i(k)$ is obtained, $\hat{p}_i(k)$ is calculated by

$$\hat{p}_i(k) = \tilde{p}_i^0(k) - \hat{\theta}_i(k).$$

Therefore, we have

$$\begin{aligned}
&\Pr\big\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k}\big\} \\
&= \Pr\big\{|\hat{\theta}_i(k) - \theta_i(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k}\big\}, \\
&= \Pr\big\{\theta_i(k) \in [\hat{\theta}_i(k) - \alpha_k, \hat{\theta}_i(k) + \alpha_k] \big| \mathcal{I}_i^{t_k}\big\} \\
&= \int_{\hat{\theta}_i(k)-\alpha_k}^{\hat{\theta}_i(k)+\alpha_k} f_{\theta_i(k)}(y|\mathcal{I}_i^{t_k})\mathrm{d}y \\
&\le \max_{\nu\in\Theta} \int_{\nu-\alpha_k}^{\nu+\alpha_k} f_{\theta_i(k)}(y)\mathrm{d}y,
\end{aligned} \tag{28}$$

where $\hat{\theta}_i(k) \in \Theta$. However, if the adversaries can only access part of $I_i^{t_k-1}$, they are unable to calculate $x_i^{t_k}(k)$ by (12) and then recover $\tilde{p}_i^0(k)$ by (27). Note that

$$x_i^{t_k+}(k) = x_i^{t_k}(k) + \tilde{p}_i^0(k) = x_i^{t_k}(k) + \theta_i(k) + p_i^0(k).$$

Hence, in this case, they need to obtain an estimation $\hat{\eta}_i(k)$ of $x_i^{t_k}(k) + \theta_i(k)$ first, and then calculate $\hat{p}_i(k)$ by

$$\hat{p}_i(k) = x_i^{t_k+}(k) - \hat{\eta}_i(k).$$

According to (12), $x_i^{t_k}(k)$ is a linear combination of the states $x_j^{(t_k-1)+}$ for $j \in \mathcal{N}_i^{\text{in},t_k-1}$. These states are dependent on some $\tilde{p}_l^0(k)$ and thus also dependent on some $\theta_l(k)$, where $l \in \mathcal{V}$. Note that the adversaries only have partial knowledge of I_i^{in,t_k-1} and know part of these states. Hence, there exist certain independent random variables, i.e., $\theta_l(k)$, of which the adversaries do not own any prior or relevant knowledge. As a result, by (22), it is hard to estimate $x_i^{t_k}(k)$ with high precision. It follows that

$$\begin{aligned}
&\Pr\big\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k}\big\} \\
&= \Pr\big\{\big|\hat{\eta}_i(k) - (x_i^{t_k}(k) + \theta_i(k))\big| \le \alpha_k \big| \mathcal{I}_i^{t_k}\big\} \\
&\le \Pr\big\{\hat{\eta}_i(k) - x_i^{t_k}(k) \in [\theta_i(k) - \alpha_k, \theta_i(k) + \alpha_k] \big| \mathcal{I}_i^{t_k}, \theta_i(k)\big\} \\
&\le \gamma,
\end{aligned} \tag{29}$$

Combining (28) and (29) together, we have

$$\begin{aligned}
&\Pr\big\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k}\big\} \\
&\quad \le p \max_{\nu\in\Theta} \int_{\nu-\alpha_k}^{\nu+\alpha_k} f_{\theta_i(k)}(y)\mathrm{d}y + \gamma \\
&\quad \triangleq h_i(\alpha_k).
\end{aligned} \tag{30}$$

At time $t > t_k$, the adversaries can estimate $p_i^0(k)$ either by the same rule that is adopted at time $t = t_k$, or by the new rule based on the new information. In the former case, we still obtain (30). We now discuss the latter case in detail. We first consider the time $t = t_k + 1$. Note that

$$\begin{aligned}
\frac{x_i^{(t_k+1)+}(k)}{a_{ii}^{t_k}} &= \frac{x_i^{t_k+1}(k)}{a_{ii}^{t_k}} \\
&= x_i^{t_k+}(k) + \frac{1}{a_{ii}^{t_k}}\Big(\sum_{j\in\mathcal{N}_i^{\text{in},t_k}\setminus\{i\}} a_{ij}^{t_k} x_j^{t_k+}(k) - \tau_{i,t_k+1}(k)\Big) \\
&= p_i^0(k) + \theta_i(k) + x_i^{t_k}(k) \\
&\quad + \frac{1}{a_{ii}^{t_k}}\Big(\sum_{j\in\mathcal{N}_i^{\text{in},t_k}\setminus\{i\}} a_{ij}^{t_k} x_j^{t_k+}(k) - \tau_{i,t_k+1}(k)\Big) \\
&= p_i^0(k) + \theta_i(k) + \theta_i'(k),
\end{aligned} \tag{31}$$

where $\tau_{i,t}(k) = \zeta_i(k)$ if $t \in \mathbb{X}_{i,k}$, i.e., when the noises are subtracted, and $\tau_{i,t}(k) = 0$ otherwise. If the full knowledge of I_i^{in,t_k} is available, the adversaries can not only collect all the $x_j^{t_k+}$ for $j \in \mathcal{N}_i^{\text{in},t}$, but also accurately infer $\tau_{i,t_k+1}(k)$ by

$$\tau_{i,t_k+1}(k) = \sum_{j\in\mathcal{N}_i^{\text{in},t_k}} a_{ij}^{t_k} x_j^{t_k+}(k) - x_i^{(t_k+1)+}(k).$$

Hence, $\theta_i'(k)$ is a deterministic constant. In this case, by using (31), we still have

$$\begin{aligned}
&\Pr\big\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k+1}\big\} \\
&\quad = \Pr\big\{|\hat{\theta}_i(k) - \theta_i(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k+1}\big\}.
\end{aligned}$$

Next, we analyze the disclosure probability of $\theta_i(k)$ given $\mathcal{I}_i^{t_k+1}$. The newly available information, i.e., the subtracted noise $\zeta_i(k)$, allows for another means of inferring $\theta_i(k)$. We now show that the resulting disclosure probability is rather small when L is drawn from an unknown distribution. Note that $\zeta_i(k) = \theta_i(k)/L > \alpha_k$. Hence,

$$\begin{aligned}
&\Pr\big\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k+1}\big\} \\
&\quad = \Pr\big\{|\hat{\theta}_i(k) - \theta_i(k)| \le \alpha_k \big| \zeta_i(k)\big\} \\
&\quad = \Pr\big\{|\hat{L} - L| \cdot \zeta_i(k) \le \alpha_k \big| \zeta_i(k)\big\} \\
&\quad = \Pr\big\{\hat{L} = L \big| \zeta_i(k)\big\} \\
&\quad \le \gamma,
\end{aligned}$$

where $\hat{L}$ is an estimation of L, and the last inequality follows from (22). Thus, the disclosure probability will not exceed the upper bound in (28), i.e.,

$$\begin{aligned}
&\Pr\big\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k+1}\big\} \\
&\quad = \Pr\big\{|\hat{\theta}_i(k) - \theta_i(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k+1}\big\} \\
&\quad \le \max_{\nu\in\Theta} \int_{\nu-\alpha_k}^{\nu+\alpha_k} f_{\theta_i(k)}(y)\mathrm{d}y.
\end{aligned} \tag{32}$$

If the full knowledge of $I_i^{\text{in},t}$ is unavailable, then $\theta_i'(k)$ contains those independent random variables whose relevant information is unknown to the adversaries. Specifically, if $t_k + 1 \le K_1$, then those variables refer to certain added noises $\theta_l(k)$ that are included in $x_l^{t_k+}(k)$, where $l \in \mathcal{V}$. Else, those variables refer to certain subtracted noises $\zeta_l(k)$ for some $l \in \mathcal{V}$. Thus, it follows from (22) that

$$\Pr\big\{|\hat{p}_i(k) - p_i^0(k)| \le \alpha_k \big| \mathcal{I}_i^{t_k+1}\big\} \le \gamma. \tag{33}$$

Combing (32) and (33), we have

$$\begin{aligned}&\Pr\bigl\{|\hat{p}_i(k)-p_i^0(k)|\le\alpha_k\big|\mathcal{I}_i^{t_k+1}\bigr\}\\ &\le p\max_{\nu\in\Theta}\int_{\nu-\alpha_k}^{\nu+\alpha_k} f_{\theta_i(k)}(y)\mathrm{d}y+\gamma\\ &=h_i(\alpha_k).\end{aligned}\tag{34}$$

A similar analysis can be performed for other arbitrary $t \geq t_k+1, t \in \mathbb{N}$. However, for $t \geq K_2$, there exists an extreme case where the adversaries successfully obtain the full knowledge of $I_i^{\mathrm{in},t}$ starting from time $t=K_1+1$ to time $t=K_2$. In this case, they can acquire $\tau_{i,t}(k)$ and perfectly infer $\theta_i(k)$ by

$$\theta_i(k)=\sum_{t=K_1+1}^{K_2}\tau_{i,t}(k).$$

Hence, the exact value of $p_i^0(k)$ can be inferred, and

$$\begin{aligned}&\Pr\bigl\{|\hat{p}_i(k)-p_i^0(k)|\le\alpha_k\big|\mathcal{I}_i^{K_2}\bigr\}\\ &=\Pr\left\{|\hat{\theta}_i(k)-\theta_i(k)|\le\alpha_k\big|\mathcal{I}_i^{K_2}\right\}\\ &=1.\end{aligned}$$

The probability that such an extreme case happens is not more than $p^{K_2-K_1}$. Therefore, we have

$$\Pr\left\{|\hat{p}_i(k)-p_i^0(k)|\le\alpha_k\big|\mathcal{I}_i^{t}\right\}\le\beta_k,$$

where

$$\beta_k=\bigl(1-p^{K_2-K_1}\bigr)h_i(\alpha_k)+p^{K_2-K_1}\tag{35}$$

for any $k=1,\ldots,m_i+1$ and $t\in\mathbb{N}$. It is easy to verify that β_k is larger than the RHS of (30). It follows that

$$\begin{aligned}&\Pr\bigl\{\|\hat{p}_i-p_i^0\|_1\le\alpha|\mathcal{I}\bigr\}\\ &=\prod_{k=1}^{m_i+1}\Pr\left\{|\hat{p}_i(k)-p_i^0(k)|\le\alpha_k\big|\mathcal{I}_i^{t}\right\}\\ &\le\prod_{k=1}^{m_i+1}\beta_k=\beta,\end{aligned}\tag{36}$$

where β is given by (24). □

References

[1] Z. He, J. He, C. Chen, and X. Guan, "Constrained distributed nonconvex optimization over time-varying directed graphs," in *Proc. 59th IEEE Conf. Decis. Control*, 2020, pp. 378–383.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[3] T. Yang, J. Lu, D. Wu, J. Wu, G. Shi, Z. Meng, and K. H. Johansson, "A distributed algorithm for economic dispatch over time-varying directed networks with delays," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5095–5106, 2016.

[4] S. Han, U. Topcu, and G. J. Pappas, "Differentially private distributed constrained optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 50–64, 2017.

[5] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, 2009.

[6] 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.

[7] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Trans. Autom. Control*, vol. 57, no. 3, pp. 592–606, 2011.

[8] K. Scaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié, "Optimal algorithms for smooth and strongly convex distributed optimization in networks," in *Proc. ICML*, 2017, pp. 3027–3036.

[9] M. T. Hale and M. Egerstedt, "Cloud-enabled differentially private multiagent optimization with constraints," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 4, pp. 1693–1706, 2018.

[10] E. Nozari, P. Tallapragada, and J. Cortés, "Differentially private distributed convex optimization via functional perturbation," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 1, pp. 395–408, 2018.

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

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

[13] S. Pu, W. Shi, J. Xu, and A. Nedic, "Push-pull gradient methods for distributed optimization in networks," *IEEE Trans. Autom. Control*, vol. 66, no. 1, pp. 1–16, 2021.

[14] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Convergence of asynchronous distributed gradient methods over stochastic networks," *IEEE Trans. Autom. Control*, vol. 63, no. 2, pp. 434–448, 2018.

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

[16] Y. Tian, Y. Sun, and G. Scutari, "Achieving linear convergence in distributed asynchronous multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 65, no. 12, pp. 5264–5279, 2020.

[17] Z. He, J. He, C. Chen, and X. Guan, "Distributed nonconvex optimization: Oracle-free iterations and globally optimal solution," *arXiv preprint arXiv:2008.00252*, 2020.

[18] Y. Mo and R. M. Murray, "Privacy preserving average consensus," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 753–765, 2017.

[19] E. Nozari, P. Tallapragada, and J. Cortés, "Differentially private average consensus: Obstructions, trade-offs, and optimal algorithm design," *Automatica*, vol. 81, pp. 221–231, 2017.

[20] J. He, L. Cai, P. Cheng, J. Pan, and L. Shi, "Consensus-based data-privacy preserving data aggregation," *IEEE Trans. Autom. Control*, vol. 64, no. 12, pp. 5222–5229, 2019.

[21] Y. Wang, "Privacy-preserving average consensus via state decomposition," *IEEE Trans. Autom. Control*, vol. 64, no. 11, pp. 4711–4716, 2019.

[22] J. He, L. Cai, and X. Guan, "Preserving data-privacy with added noises: Optimal estimation and privacy analysis," *IEEE Trans. Inf. Theory*, vol. 64, no. 8, pp. 5677–5690, 2018.

[23] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. 44th Annu. IEEE Symp. Found. Comput. Sci.*, 2003, pp. 482–491.

[24] A. Nedić, A. Olshevsky, and M. G. Rabbat, "Network topology and communication-computation tradeoffs in decentralized optimization," *Proc. IEEE*, vol. 106, no. 5, pp. 953–976, 2018.

[25] R. O. Saber and R. M. Murray, "Consensus protocols for networks of dynamic agents," in *Proc. Amer. Control Conf.*, 2003, pp. 951–956.

[26] L. N. Trefethen, *Approximation theory and approximation practice*. SIAM, 2013, vol. 128.

[27] J. He, L. Cai, C. Zhao, P. Cheng, and X. Guan, "Privacy-preserving average consensus: privacy analysis and algorithm design," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 1, pp. 127–138, 2019.

[28] J. P. Boyd, *Solving Transcendental Equations: The Chebyshev Polynomial Proxy and Other Numerical Rootfinders, Perturbation Series, and Oracles*. SIAM, 2014, vol. 139.

[29] M. Prakash, S. Talukdar, S. Attree, S. Patel, and M. V. Salapaka, "Distributed stopping criterion for ratio consensus," in *Proc. 56th Annu. Allerton Conf. on Commun., Control and Computing*, 2018, pp. 131–135.

[30] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[31] G. Blekherman, P. A. Parrilo, and R. R. Thomas, *Semidefinite optimization and convex algebraic geometry*. SIAM, 2013, vol. 13.

[32] G. Scutari and Y. Sun, "Distributed nonconvex constrained optimization over time-varying digraphs," *Math. Program.*, vol. 176, no. 1-2, pp. 497–544, 2019.

[33] J. He, L. Cai, and X. Guan, "Differential private noise adding mechanism and its application on consensus algorithm," *IEEE Trans. Signal Process.*, vol. 68, pp. 4069–4082, 2020.

[34] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.

[35] C. N. Hadjicostis and T. Charalambous, "Average consensus in the presence of delays in directed graph topologies," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 763–768, 2013.

[36] N. Bof, R. Carli, and L. Schenato, "Average consensus with asynchronous updates and unreliable communication," in *Proc. IFAC World Congr.*, 2017, pp. 601–606.

[37] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1245–1260, 2018.

[38] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the linear convergence of the ADMM in decentralized consensus optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1750–1761, 2014.

[39] P. Di Lorenzo and G. Scutari, "NEXT: In-network nonconvex optimization," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 2, pp. 120–136, 2016.

[40] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.

[41] P. Bianchi and J. Jakubowicz, "Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization," *IEEE Trans. Autom. Control*, vol. 58, no. 2, pp. 391–405, 2012.

[42] T. Tatarenko and B. Touri, "Non-convex distributed optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3744–3757, 2017.

[43] M. Hong, D. Hajinezhad, and M.-M. Zhao, "Prox-PDA: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks," in *Proc. ICML*, 2017, pp. 1529–1538.

[44] Z. Huang, S. Mitra, and G. Dullerud, "Differentially private iterative synchronous consensus," in *Proc. ACM Workshop Privacy Electron. Soc.*, 2012, pp. 81–90.

[45] M. Ruan, H. Gao, and Y. Wang, "Secure and privacy-preserving consensus," *IEEE Trans. Autom. Control*, vol. 64, no. 10, pp. 4035–4049, 2019.

[46] C. N. Hadjicostis and A. D. Dominguez-Garcia, "Privacy-preserving distributed averaging via homomorphically encrypted ratio consensus," *IEEE Trans. Autom. Control*, vol. 65, no. 9, pp. 3887–3894, 2020.

[47] Y. Lu and M. Zhu, "Privacy preserving distributed optimization using homomorphic encryption," *Automatica*, vol. 96, pp. 314–325, 2018.

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

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

[50] A. Agarwal and J. C. Duchi, "Distributed delayed stochastic optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 873–881.