Privacy-Preserving Incremental ADMM for Decentralized Consensus Optimization

Yu Ye, Student Member, IEEE, Hao Chen, Student Member, IEEE, Ming Xiao, Senior Member, IEEE, Mikael Skoglund, Fellow, IEEE, and H. Vincent Poor, Fellow, IEEE

Abstract—The alternating direction method of multipliers (ADMM) has been recently recognized as a promising optimizer for large-scale machine learning models. However, there are very few results studying ADMM from the aspect of communication costs, especially jointly with privacy preservation, which are critical for distributed learning. We investigate the communication efficiency and privacy-preservation of ADMM by solving the consensus optimization problem over decentralized networks. Since walk algorithms can reduce communication load, we first propose incremental ADMM (I-ADMM) based on the walk algorithm, the updating order of which follows a Hamiltonian cycle instead. However, I-ADMM cannot guarantee the privacy for agents against external eavesdroppers even if the randomized initialization is applied. To protect privacy for agents, we then propose two privacy-preserving incremental ADMM algorithms, i.e., PI-ADMM1 and PI-ADMM2, where perturbation over step sizes and primal variables is adopted, respectively. Through theoretical analyses, we prove the convergence and privacy preservation for PI-ADMM1, which are further supported by numerical experiments. Besides, simulations demonstrate that the proposed PI-ADMM1 and PI-ADMM2 algorithms are communication efficient compared with state-of-the-art methods.

Index Terms—Decentralized optimization, alternating direction method of multipliers (ADMM), privacy preservation.

I. INTRODUCTION

Consider a decentralized network \( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \), where \( \mathcal{V} = \{1, \ldots, N\} \) is the set of agents with computing capability, and \( \mathcal{E} \) is the set of links. \( \mathcal{G} \) is connected and there is no center agent. The agents seek to solve the consensus optimization problem (1) collaboratively through sharing information among each other,

\[
\min_x \sum_{i=1}^N f_i(x; \mathcal{D}_i),
\]

where \( f_i : \mathbb{R}^p \to \mathbb{R} \) is the local function at agent \( i \), and \( \mathcal{D}_i \) is the private dataset at agent \( i \). All agents share a common optimization variable \( x \in \mathbb{R}^p \). The decentralized optimization problem (1) is widely applied in many areas such as signal processing [1]–[3], machine learning [4]–[6], wireless sensor networks (WSNs) [7], [8] and smart grids [9]–[11], just to name a few.

In the existing literature, e.g., [11]–[18], a large number of decentralized algorithms have been investigated to solve the consensus problem (1). Typically, the algorithms can mainly be classified into primal and primal-dual types, namely, gradient descent based (GD) methods and the alternating direction method of multipliers (ADMM) based methods, respectively. In this work, we will use ADMM as the optimizer, which can usually achieve more accurate consensus performance than GD based methods with constant step size [18].

Though distributed gradient descent (DGD) [12], EXTRA [13] and distributed ADMM (D-ADMM) [14] have good convergence rates with respect to the number of iterations, these algorithms require each agent to collect information from all its neighbors. This makes the amount of communication high for each iteration. Hence, for the applications with unstable links among agents such as WSNs, the communication load becomes one of the main bottlenecks. Moreover the straggler problem is also pronounced in synchronous ADMM based methods [16], where the total running time is significantly increased due to slowly computing nodes. Besides, for the ADMM based decentralized approaches provided in [14]–[18], agents are required to exchange primal and dual variables with neighbors in each iteration. This inevitably leads to the privacy problem if the local function \( f_i \) and data set \( \mathcal{D}_i \) are private to agent \( i \in \mathcal{N} \), for instance, in the practical applications with sensitive information such as personal medical or financial records, confidential business notes and so on. Thus, guaranteeing privacy is critical for decentralized consensus algorithms.

A. Related Works

To reduce the communication load for solving consensus problem (1), various decentralized approaches have been proposed recently. Among these efforts, one important direction...
is to limit information sharing for each iteration. In [19], given an underlying graph, the weighted ADMM is developed by deleting some links prior to the optimization process. Communication-censored ADMM (COCA) in [20] can adaptively determine whether a message is informative during the optimization process. Following COCA, an extreme scenario is to only activate one link in the graph per iteration such as the random-walk ADMM (W-ADMM) [17], which incrementally updates the optimization variables. To achieve optimized trade-off between communication cost and running time, parallel random walk ADMM method (PW-ADMM) and intelligent PW-ADMM (IPW-ADMM) are proposed in [18]. References [21], [22] also take advantages of random walks but deal with stochastic gradients. Another line of works is to exchange sparse or quantized messages to transmit fewer bits. Following this direction, the quantized ADMM is provided in [23]. Two quantized stochastic gradient descent (SGD) methods, sign SGD with error-feedback (EF-SignSGD) and periodic quantized averaging SGD (PQASGD), are proposed in [24] and [25], respectively. Aji et al. [26] presented a heuristic approach to truncate the smallest gradient components and only communicate the remaining large ones. However, these algorithms cannot guarantee exact convergence to the optimal solution [27].

Meanwhile, with increasing concerns on data privacy, e.g., GDPR in Europe [28], more and more efforts are allocated to developing privacy preserving algorithms for solving the decentralized consensus problem (1). To measure the privacy preserving performance, differential privacy has been used [29]. In general, artificial uncertainty is introduced over shared messages or local functions to maintain sensitive information secure against eavesdroppers. Differential private distributed algorithms based on GD have been presented in [30]–[32], where noise following different distributions is added to variable or gradients. Moreover, in primal-dual methods [33]–[36], perturbation is conducted either on penalty term, or on the primal and dual variables before sharing to neighboring agents. On the other hand, cryptography based methods, such as (partial) homomorphic encryption [37]–[39], have also been adopted to protect the privacy of agents through applying the encryption mechanism to exchanged information. Other privacy-preserving approaches, including perturbing problems or states, are provided in [40], [41]. However, these methods lead to high computational overheads and communication load especially in large-scale decentralized optimization.

B. Motivation and Contributions

W-ADMM can significantly reduce communication load by activating only one node and link per iteration in a successive manner. Since the sequence of the updating order for agents is randomized by following a Markov chain, it is possible that the primal and dual variables at some agents are updated for much fewer rounds than others. Thus, the convergence speed may degrade. To guarantee agents not to be inactive for a long time, Random Walk with Choice (RWC) is introduced in IPW-ADMM [18] to reduce running time. The main principle of RWC is that through restricting the updating order of agents, the convergence speed can be improved for incremental approaches, which shall save communication load. This phenomenon is also supported by the walk proximal gradient (WPG) presented in [11], where the updating order of agents follows a Hamiltonian Cycle. Therefore, in what follows, we will fix the updating order of agents as WPG and present the incremental ADMM (I-ADMM). Different from WPG, the I-ADMM is a primal-dual based approach. Moreover, comparing to PW-ADMM and IPW-ADMM, there is only one walk with a fixed order in I-ADMM.

To enable the privacy preservation for I-ADMM against the eavesdropper, which overhears all links in $\mathcal{E}$, we introduce uncertainty among local primal-dual variables and the transmitted tokens, e.g., through adding artificial noise over primal or dual variables [33], [35]. However, without scaling noise with iterations, there exists an error bound at convergence [35], which leads to a trade-off between privacy and accuracy. To avoid compromising convergence performance, we propose to perform perturbation over the step size for both updates of primal and dual variables, i.e., $\{x_i, y_i\} \in \mathcal{V}$. Equivalently, this procedure can be seen as adding noise, which is scaled by the primal and dual residues, over $x^k_i$ and $y^k_i$, respectively for iteration $k$.

Different from the existed results, in what follows, we will study the ADMM based optimizer for solving (1) from the aspect of both communication efficiency and privacy preservation. The main contributions of this article can be summarized as follows.

• We propose the I-ADMM method for solving decentralized consensus optimization (1). Different from other incremental ADMM methods such as W-ADMM [17], the update order of I-ADMM follows a Hamiltonian cycle. Through theoretical analysis, we show that I-ADMM exhibits linear convergence rate for solving decentralized least-squares problems, and we demonstrate improved communication efficiency compared to W-ADMM.

• Since I-ADMM is not privacy-preserving, we first introduce the randomized initialization. To guarantee privacy preservation, we provide two privacy-preserving incremental ADMM (PI-ADMM) algorithms, i.e., PI-ADMM1 and PI-ADMM2, which apply perturbation over step size and primal variable, respectively. To the best of our knowledge, this is the first article to consider privacy preservation for incremental ADMM.

• We prove the convergence of PI-ADMM1, and show that at convergence, the primal and dual variables generated by PI-ADMM1 satisfy the Karush-Kuhn-Tucker (KKT) conditions with mild assumptions over local functions. In addition, we prove that privacy preservation is guaranteed against honest-but-curious colluding agents or external eavesdroppers.

• Numerical results show that the proposed I-ADMM based algorithms are communication efficient compared with state-of-the-art methods. Moreover, we show that PI-ADMM1 can guarantee both privacy-preservation and accurate convergence.

The remainder of this article is organized as follows. We first introduce the incremental ADMM in Section II. To guarantee the privacy of local agents against external eavesdroppers, we present two PI-ADMM algorithms in Section III. The convergence and privacy analyses regarding proposed approaches are presented in Section IV. To validate the efficiency of proposed
methods, we provide numerical experiments in Section V. Finally, Section VI concludes the article.

II. INCREMENTAL ADMM

By defining $x = [x_1, ..., x_N] \in \mathbb{R}^{PN}$, problem (1) can be rewritten as

$$\min_{x,z} \sum_{i=1}^{N} f_i(x_i; D_i), \text{s.t.} \bigoplus z - x = 0,$$

where $z \in \mathbb{R}^p$, $1 = [1, ..., 1] \in \mathbb{R}^N$, and $\bigoplus$ is Kronecker product. For simplicity, we denote $f'_i(x_i; D_i)$ as $f_i(x_i)$. The augmented Lagrangian for problem (2) is

$$L_\rho(x, y, z) = \sum_{i=1}^{N} f_i(x_i) + \langle y, \bigoplus z - x \rangle + \frac{\rho}{2} \| \bigoplus z - x \|^2,$$

where $y = [y_1, ..., y_N] \in \mathbb{R}^{PN}$ is the dual variable, while $\rho > 0$ is a constant parameter. Following W-ADMM [17], with guaranteeing $\sum_{i=1}^{N} (x'_i - x'_i) = 0$, the updates of $x$, $y$ and $z$ at the $(k+1)$-th iteration are given by

$$x'^{k+1} = \left\{ \begin{array}{ll}
\arg \min_{x_i} f_i(x_i) + \frac{\rho}{2} \| z^k - x_i + \frac{y'^k}{\rho} \|^2, & i = i_k; \\
\bigoplus_{i \neq i_k} x_i^k, & i \notin \mathcal{V}, \end{array} \right.$$

$$y'^{k+1} : = \left\{ \begin{array}{ll}
y'^k + \rho (z^k - x'^{k+1}), & i = i_k; \\
y'^k, & i \notin \mathcal{V}, \end{array} \right.$$

$$z^{k+1} = \frac{1}{N} \sum_{i=1}^{N} \left( x'^{k+1} - \frac{y'^{k+1}}{\rho} \right).$$

Note that the updates (4a)–(4c) are not decentralized since (4c) collects information from all agents. With initializing $x^0 = 0$ and $y^0 = 0$, it is easy to verify that the update of $z$ can be incrementally implemented as

$$z^{k+1} = z^k + \frac{1}{N} \left( x'^{k+1} - \frac{y'^{k+1}}{\rho} - \left( x_k^k - \frac{y_k^k}{\rho} \right) \right).$$

Then the decentralized implementation of I-WADMM is presented in Algorithm 1. Different from W-ADMM, the agents are activated in a predetermined circulant pattern for I-ADMM, where the activated agent in the $k$-th iteration is $i = k \mod N + 1$. Furthermore, we assume walk $(i_k)_{k \geq 0}$ repeats a Hamiltonian cycle order $1 \rightarrow 2 \rightarrow \cdots \rightarrow N \rightarrow 1 \rightarrow \cdots$ as WPG [11]. There are many existing works studying how to find a Hamiltonian cycle in a graph [42]–[44]. The authors in [44] prove that Hamiltonian cycle in an undirected graph can be found in polynomial time. Many graphs, including complete graphs, ring graphs, and 4-connected planar graph have been proven to contain a Hamiltonian cycle. Since we focus on investigating the communication cost consumed by iterative updates in I-ADMM, we omit the complexity for finding the Hamiltonian cycle in $G$.

In I-ADMM, the variable $z^{k+1}$ gets updated at agent $i_k$ and passed as to its neighbor $i_{k+1}$ through Hamiltonian cycle. In Fig. 2, we present the evolution of $\{x_i, y_i\}$ at agents and the transition of token $z$ when I-ADMM is applied to the example shown in Fig. 1.

Algorithm 1: I-ADMM.

1: initialize: $\{z^0 = 0, x_i^0 = 0, y_i^0 = 0, k = 0 | i \in \mathcal{V}\}$
2: for $k = 0, 1, \ldots$ do
3: agent $i_k = k \mod N + 1$ do:
4: receive token $z^k$;
5: update $x^{k+1}$ according to (4a);
6: update $y^{k+1}$ according to (4b);
7: update $z^{k+1}$ according to (5);
8: send token $z^{k+1}$ to agent $i_{k+1} = k \mod N + 2$;
end for

A. Convergence Analysis

To start with the convergence analysis for I-ADMM, we first make the following assumptions over network $G$ and local functions $\{f_i\}$. For simplicity, we denote $F(x) = \sum_{i=1}^{N} f_i(x)$.

Assumption 1: Graph $G$ is well connected and there exists a Hamiltonian cycle.

Assumption 2: Each local loss function $f_i(x)$ is bounded from below over $x$, and $f_i(x)$ is coercive over $x$. Each $f_i(x)$ is $L$-Lipschitz differentiable, i.e., for any $u, v \in \mathbb{R}^p$,

$$\| \nabla f_i(u) - \nabla f_i(v) \| \leq L \| u - v \|, i = 1, \ldots, N.$$

Then according to Assumption 2 and Theorem 1 in [17], we have the following convergence property for I-ADMM.

1 All the I-ADMM based algorithms presented in this article can also be applied to the non-Hamiltonian graph with the shortest path cycle [11], where the provided convergence (except Remarks 1 and 3) and privacy analysis still hold.
Lemma 1: Under Assumptions 1 and 2, for $\rho \geq 2L + 2$, the iterations $(x^k, y^k, z^k)$ generated by I-ADMM satisfy the following properties:

1. $\mathcal{L}_\rho(x^k, y^k, z^k) - \mathcal{L}_\rho(x^{k+1}, y^{k+1}, z^{k+1}) \geq 0$;
2. $(\mathcal{L}_\rho(x^k, y^k, z^k))_{k \geq 0}$ is lower bounded and convergent;
3. $\lim_{k \to \infty} ||\nabla \mathcal{L}_\rho(x^k, y^k, z^k)|| = 0$.

Proof: The convergence proof of I-ADMM is similar to that of W-ADMM in [17]. The difference is that as Assumption 1 in [17], each agent is visited with a fixed period, i.e., $\mathcal{F}(\delta) = N$.

To analyze the convergence rate and communication complexity, we focus on the decentralized least-squares problem (21) in [17]. Then we have the following results.

Remark 1: I-ADMM has linear convergence rate for solving decentralized least-squares problems, where the communication complexity is roughly

$$O \left( \ln \left( \frac{N}{\epsilon} \right) / \ln \left( 1 + \frac{1}{4\rho^2 N} \right) \right) \approx O \left( \frac{1}{\epsilon} N \ln(N) \right),$$

with $\epsilon$ being a target accuracy. I-ADMM can be more communication efficient than W-ADMM over graphs where a Hamiltonian cycle exists.

Proof: To prove the linear convergence rate and communication complexity of I-ADMM, one can follow the proof of Theorem 3 and communication analysis in [17]. Recalling the communication complexity of W-ADMM as

$$O \left( \ln \left( \frac{1}{\epsilon} \right) N \ln^3(N) \right),$$

where $P$ is the transition probability matrix for token $z$, and $\lambda_2(P) = \max \{|\lambda_i(P)| | \lambda_i(P) \neq 1\}$ is the eigenvalue of $P$. Due to $(1 - \lambda_2(P))^2 \in (0, 1]$, we can conclude I-ADMM is more communication efficient than W-ADMM.

B. Computation Efficient I-ADMM

With complex local function $f_i$, the computation load for solving minimization problem (4a) is high. To alleviate computation burden for the update of $x^{k+1}$ in I-ADMM, we can apply first-order approximation by replacing (4a) with update:

$$x^{k+1}_i := \begin{cases} z^k + \frac{y^k_i}{\rho} - \frac{1}{\rho} \nabla f_i(x^k_i), & i = i_k; \\ x^k_i, & i \neq i_k. \end{cases}$$

Comparing to (4a), update process (9) can save computation. However, it may cause higher communication loads to achieve the same accuracy, as shown in [45]. In what follows, we will focus on investigating the privacy preserving I-ADMM based on Algorithm 1. Besides, the obtained results can also be extended to the computation efficient I-ADMM methods with privacy preservation.

III. INCREMENTAL ADMM WITH PRIVACY PRESERVATION

In this section, we will first analyze the privacy of Algorithm 1. Then to guarantee the security for agents in solving (2), we propose the PI-ADMM algorithms.

A. Review of I-ADMM

Assume that there exists an external eavesdropper overhearing all the links, as shown in Fig. 1. The agent wants to protect its private function against honest-but-curious adversaries and external eavesdroppers, which are commonly used attack models in privacy studies [33], [46]. The initialization and updating rule of algorithms are assumed to be publicly known to adversaries and eavesdroppers [33]. Then the privacy property of I-ADMM is analyzed as follows.

Lemma 2: In Algorithm 1, the intermediate states and gradients $\{x^{k+1}_i, y^{k+1}_i, \nabla f_i(x^k_i)| i \in \mathcal{V}, k = 0, 1, \ldots\}$ can be inferred.

Proof. To prove this lemma, we first assume that $x^{k+1}_i$ and $y^{k+1}_i$ at the end of iteration $k(\geq 0)$ are known to the external eavesdropper. Since the initialization and updating rule in algorithms are publicly known, the external eavesdropper can establish the following equations with the update process (4a):

$$\nabla f_{i_k}(x^{k+1}_{i_k}) = y^{k+1}_{i_k}. \tag{10}$$

Hence once $y^{k+1}_{i_k}$ is inferred by the master, $\nabla f_{i_k}(x^{k+1}_{i_k})$ will be revealed as well. Combining the update process (4b) and (5), we have

$$\begin{cases} y^{k+1}_{i_k} = y^k_{i_k} + \rho (z^k - x^{k+1}_{i_k}); \\ \Delta^{k+1} = \frac{1}{N} \left[ (x^{k+1}_{i_k} - y^{k+1}_{i_k}) - \left( \frac{y^k_{i_k} - y^{k+1}_{i_k}}{\rho} \right) \right], \end{cases} \tag{11}$$

where $\Delta^{k+1} = z^{k+1} - z^k$. Then through rearranging (11), the recursive equations are given by

$$\begin{cases} x^{k+1}_{i_k} = \frac{1}{2} (N\Delta^{k+1} + z^k + x^k_{i_k}); \\ y^{k+1}_{i_k} = y^k_{i_k} + \frac{\rho}{2} (z^k - N\Delta^{k+1} - x^k_{i_k}). \end{cases} \tag{12}$$

Since the eavesdropper also overhears $z^{k+1}$ and $z^k$, equation (12) consists of 2 equations with 2 variables. Hence $x^{k+1}_{i_k}$ and $y^{k+1}_{i_k}$ can be obtained by solving (12). In I-ADMM, since $\{x^0_{i_k}, y^0_{i_k}| i \in \mathcal{V}\}$ are known to the eavesdropper, with (10) and the update (4a), (4b), we can conclude that $\{x^k_{i_k}, y^k_{i_k}, \nabla f_i(x^k_i)| i \in \mathcal{V}, k = 0, 1, \ldots\}$ can also be inferred.

In addition, with $\{\nabla f_i(x^k_i)| i \in \mathcal{V}, k = 0, 1, \ldots\}$, the external eavesdropper can construct a piece-wise linear lower bound on the private function $f_i(\cdot)$, as well as the form of $f_i(\cdot)$, e.g., whether it is a quadratic, exponential, or other forms [33].

B. Privacy-Preserving I-ADMM

Inspired by the privacy-preservation definitions in [33], [47], [48], we define the privacy as follows.

Definition 1: A mechanism $M : \mathcal{X} \rightarrow \mathcal{Y}$ is defined to be privacy preserving if the input $X$ cannot be uniquely derived from the output $Y$.

This definition is based on the fact that if a system of equations has an infinite number of solutions, it is impossible to derive the exact value of the original input data from the output data, which hence establishes privacy preservation [49]. From Lemma 2, we then conclude that I-ADMM is not privacy preserving.
Recalling the proof of Lemma 2, by making \( x_0^k \) and \( y_0^k \) private, information \( \{ x_i^k, y_i^k, \nabla f_i(x_i^k) \}_{i=1}^N \) cannot be inferred by the eavesdropper exactly from the recursive equation (12) if the states and multipliers are unknown at the end of iterative algorithm. In fact, to implement the incremental update (5), we only need to ensure \( \sum_{i=1}^{N} x_i^k = 0 \).

To introduce uncertainty, we randomize the initialization and apply perturbation over step size \( \rho \) for primal and dual updates, as summarized in Algorithm 2 (PI-ADMM1). In the initialization phase, local variables \( x_i^0 \) and \( y_i^0 \) are randomly generated at agent \( i \) such that (4c) is satisfied. Then the update phase is carried out with initialization \( \{ x_i^0 = v, y_i^0 = -\rho v | i \in \mathcal{V} \} \). The primal and dual variables are updated by (13) and (14) iteratively, where the step size \( \rho \) is multiplied by a perturbation factor \( \gamma_i^k \). Each agent can select \( \mathbb{P}_i \) independently, and \( \gamma_i \) is randomly generated according to distribution \( \mathbb{P}_i \). It is worth noting that (14) can be regarded as introducing additive noise, i.e., \( \rho(\gamma_i^k - 1)(x_i^{k+1} - x_i^k) \), over the dual variable updating, which is scaled by the primal residue. Regarding to the primal update with the first order approximation, it can also be expressed as appending the scaled noise \( \frac{1}{\rho} \frac{\gamma_i^k - 1}{\rho} (y_i^{k+1} - y_i^k) \) over \( x_i^{k+1} \). Different from existing ADMM based privacy-preserving methods [29], [33], we randomly generate the initialization and randomize the primal and dual updates through the step-size perturbation in PI-ADMM1 instead of implementing encryption for each iteration. Thus, the computation complexity and communication load of PI-ADMM1 are significantly reduced. In addition, the applied perturbation mechanism avoids computation over step size \( \rho \) as Algorithm 1 in [33].

From the results of [29], another approach to enhance the privacy of PI-ADMM1 is perturbing the primal updates by adding noise according to fixed stochastic, which is illustrated in Algorithm 3 (PI-ADMM2). For the \( k \)-th iteration, an artificial noise \( \omega_i^k \) generated at agent \( i \) locally affilitated over \( x_i^{k+1} \). From the analyses in [29], this perturbation mechanism can also guarantee the privacy for agents. However, due to page limitation, we only evaluate the convergence property for PI-ADMM2 in Section V.

### IV. CONVERGENCE AND PRIVACY ANALYSIS

In this section, we will first prove the convergence for PI-ADMM1 and PI-ADMM2. Then the privacy analysis for PI-ADMM1 is provided.

A. Convergence Analysis

We first provide the convergence of PI-ADMM1 only with randomized initialization.

**Remark 2:** With \( \{ \gamma_i^k \}_{i=1}^N \) for PI-ADMM1 has the same convergence properties as those of I-ADMM presented in Lemma 1.

**Proof:** Since \( \sum_{i=1}^{N} (x_i^k - y_i^k) = 0 \) is guaranteed from steps 2-7 in PI-ADMM1, the convergence can be proved as in Lemma 1.

Denoting \( x^* = \arg \min_{x} F(x) \), then with Assumption 2, the convergence properties of PI-ADMM1 is shown as follows.

**Lemma 3:** With the following conditions

\[
\rho > L; \tag{16a}
\]

\[
\gamma_i^k > \max \left\{ \frac{2 \rho^2 + 4 \rho + 1}{\rho - L}, 2(\rho + 2)N \right\}, k = 0, 1, \ldots, \tag{16b}
\]

the iterations \( \{x^k, y^k, z^k\} \) generated by PI-ADMM1 satisfy the following properties:

1) \( \mathcal{L}_P(x^k, y^k, z^k) - \mathcal{L}_P(x^{k+1}, y^{k+1}, z^{k+1}) \geq 0; \)
2) \( \mathcal{L}_P(x^k, y^k, z^k)_{k \geq 0} \) is lower bounded by the optimum \( F(x^*) \) and convergent.

**Proof:** See Appendix A.
Based on the analysis of Lemma 3 for PI-ADMM1, the convergence property is summarized in the following theorem.

**Theorem 1:** Following Assumptions 1 and 2, and the conditions given in Lemma 3, the iterations \( (x^k, y^k, z^k) \) generated by PI-ADMM1 have limit points, which satisfies the KKT conditions of problem (2).

**Proof:** See Appendix B.

It should be noted that the conditions given in Lemma 3 and Theorem 1 is sufficient for the convergence of PI-ADMM1.

**Remark 3:** PI-ADMM1 has linear convergence rate for solving the decentralized least-squares problems, where the communication complexity is roughly \( O(\ln(1/\epsilon)N^2 \ln(N)) \) with \( \gamma = \max \{E[\gamma_i] | i \in \mathcal{V} \} \).

**Proof:** By substituting \( \rho \gamma_i \) into (7) and taking the expectation over \( \gamma_i \), we can obtain the desired result.

According to the constraint shown in Lemma 3, the available range for \( \gamma_i \) will increase with increasing \( E[\gamma_i] \). However, the communication complexity of PI-ADMM1 will also increase when \( E[\gamma_i] \) grows.

### B. Privacy Analysis

In what follows, we will analyze the privacy properties for PI-ADMM1.

**Remark 4:** With \( \{\gamma_{ik}^k = 1| k = 0, ..., K \} \), the primal and dual variables, and gradients, i.e., \( \{x_i^{k+1}, y_i^{k+1}, \nabla f_i(x_i^k) | i \in \mathcal{V}, k = 0, ..., K \} \), generated by PI-ADMM1, cannot be inferred by the eavesdropper exactly if \( z^{k+1} \neq z_k^{k+1} \) for all \( k = 0, ..., K \).

**Proof:** Assume that the eavesdropper collects information from \( K \) iterations to infer the information of agents. Then with \( \{\gamma_{ik}^k = 1| k = 0, ..., K \} \) and according to (12), the measurement considering all the agents can be formulated as

\[
\begin{align*}
\{x_i^0 - y_i^0 \rho = 0, i \in \mathcal{V} ;
\ x_i^1 = \frac{1}{2} (N \Delta^1 + z^0 + x_i^0^0) ;
\ y_i^1 = y_i^0 + \frac{\rho}{2} (z^0 - N \Delta^1 - x_i^0^0) ;
\vdots
\ x_i^{K+1} = \frac{1}{2} (N \Delta^{K+1} + z^K + x_i^{K}^0) ;
\ y_i^{K+1} = y_i^K + \frac{\rho}{2} (z^K - N \Delta^{K+1} - x_i^0^{K}) .
\end{align*}
\]

In (17), \( \{z^k| k = 0, ..., K \} \) are known to the eavesdropper, while \( \{x_i^k, y_i^k, x_i^{k+1}, y_i^{k+1}| k = 0, ..., K \} \) are unknown variables to the eavesdropper. Thus, (17) consists of 2\( K \) + 3 equations and 2\( K \) + 2\( N \) + 2 unknown variables. Since \( N > 2 \), the eavesdropper cannot solve (17) to infer the exact values of \( \{x_i^k, y_i^k | i \in \mathcal{V}, k = 0, ..., K \} \). Hence with (10), the gradients \( \{\nabla f_i(x_i^k) | i \in \mathcal{V}, k = 0, ..., K \} \) cannot be inferred exactly by the eavesdropper either.

The consensus happens when PI-ADMM1 converges, where \( z \) coincides with local variables \( \{x_i\} \). Hence the eavesdropper can use \( z \) to infer \( \{x_i\} \). However, with finite iterations, the gap between token \( z \) and states \( \{x_i\} \) can only be guaranteed within a threshold \( \epsilon^2 \). Denote \( \tilde{z} \) as the measurement of the eavesdropper regarding to \( x_i \). According to (12), once \( z_{ik}^{k+1} \) is measured, all the states \( \{x_{ik}^k | k = 0, ..., K \} \) of agent \( i_k \) can also be estimated sequentially.

**Remark 5:** With \( \{\gamma_{ik}^k = 1 | k = 0, ..., K \} \) in PI-ADMM1, and the eavesdropper using \( z_{ik}^{K+1} \) to infer \( x_{ik}^{K+1} \) of agent \( i_k, K \) mod \( N + 1 \), if \( \|z_{ik}^{K+1} - x_{ik}^{K+1}\| < \epsilon \), there exists an error bound for measurement (17) as

\[
\|\tilde{z}_i^k - z_i^k\| < 2\epsilon^2, \left\|y_i^k - y_i^{K} \right\| < \rho \left( 2\left(\frac{\epsilon}{\sqrt{n}}\right)^2 - 2\epsilon \right),
\]

where \( k \in [K - nN + 1, K - (n - 1)N], 1 \leq n \leq \left\lfloor \frac{K}{N} \right\rfloor \).

**Proof:** See Appendix C.

**Remark 6:** With \( \{\gamma_{ik}^k = 1 | k = 0, 1, \ldots \} \), the primal and dual variables, and gradients, i.e., \( \{x_i^k, y_i^k, \nabla f_i(x_i^k) | i \in \mathcal{V}, k = 0, ..., K \} \), generated by PI-ADMM1, can be inferred by the eavesdropper asymptotically, i.e., as \( K \to \infty \). According to [17], it is guaranteed that \( \lim_{K \to \infty} z_i^k = x_i^k, \forall i \in \mathcal{V} \). Furthermore, with equation (12) and condition \( x_i^0 - y_i^0 \rho = 0 \), the values \( \{x_i^k, y_i^k | k = 0, ..., K \} \) can be derived recursively. Due to (10), the gradients \( \{\nabla f_i(x_i^k) | k = 0, ..., K \} \) is revealed.

Remark 6 states that only enabling randomized initialization in PI-ADMM1 cannot ensure privacy asymptotically. But with perturbations in the primal and dual updates, it can guarantee augmented privacy preservation.

**Theorem 2:** In PI-ADMM1, the exact states \( \{x_i^k | i \in \mathcal{V}, k = 0, 1, \ldots \} \) cannot be inferred by the eavesdropper unless \( z_{ik}^{k+1} = x_{ik}^{k+1} \) holds. The multipliers and gradients \( \{y_i^k, \nabla f_i(x_i^k) | i \in \mathcal{V}, k = 0, 1, \ldots \} \) cannot be exactly inferred either.

**Proof:** Due to the randomized initialization, the values of \( x_i^0 \) and \( y_i^0 \) cannot be revealed by the eavesdropper. Assume that the eavesdropper collects information from \( K \) iterations to infer the information of agents. The measurement considering all the agents can be formulated as (19).

\[
\begin{align*}
\{x_i^0 - y_i^0 \rho = 0, i \in \mathcal{V} ;
\ x_i^1 = \frac{1}{1 + \gamma_{i0}} (N \Delta^1 + \gamma_{i0} z^0 + x_i^0) ;
\ y_i^1 = y_i^0 + \frac{\rho_{i0}}{1 + \gamma_{i0}} (z^0 - N \Delta^1 - x_i^0) ;
\vdots
\ x_i^{K+1} = \frac{1}{1 + \gamma_{i0}} (N \Delta^{K+1} + \gamma_{i0} z^K + x_i^{K}) ;
\ y_i^{K+1} = y_i^K + \frac{\rho_{i0}}{1 + \gamma_{i0}} (z^K - N \Delta^{K+1} - x_i^{K}) .
\end{align*}
\]

Different form (17), in the above equations, the tokens \( \{z_{ik}^{k+1}| k = 0, \ldots, K + 1 \} \) are known to the eavesdropper, while \( \epsilon^2 \) can be a predetermined stopping criterion for PI-ADMM1.
primal and dual variables \( \{ x_k^i, y_k^i \mid i \in \mathcal{V}, k = 0, \ldots, K \} \) and perturbations \( \{ \gamma_k^i \mid k = 0, \ldots, K \} \) are unknown variables. Thus, (19) consists of \( 2K + N + 2 \) equations and \( 3K + 2N + 3 \) unknown variables. Thus the eavesdropper cannot solve (19) to infer the exact values of \( \{ x_k^i, y_k^i \mid i \in \mathcal{V}, k = 0, \ldots, K \} \) since \( K + N + 1 > 0 \). When \( K \to \infty \), the eavesdropper can have another piece of information according to the KKT conditions (34). However, due to \( K \gg N \), the undetermined system (19) still cannot be solved asymptotically.

**Corollary 1:** In PI-ADMM1, the local functions \( \{ f_i \mid i \in \mathcal{V} \} \) cannot be inferred by the eavesdropper.

**Proof:** According to Theorem 2, the states and corresponding gradients are unknown to the eavesdropper. Hence from [33], the local objective functions \( \{ f_i \mid i \in \mathcal{V} \} \), cannot be inferred by the eavesdropper.

Till now, we have assumed that the eavesdropper can overhear all the links among agents. However, in some cases (e.g., large scale networks), the external eavesdropper can only wiretap some of the links [39]. Hence it is worth analyzing the privacy properties of PI-ADMM1 in such settings.

**Remark 7:** When the external eavesdropper overhears links \( \mathcal{V} \subseteq \mathcal{V} \) with PI-ADMM1, the dual variables, gradients and local loss functions, i.e., \( \{ y_k^i, \nabla f_i(x_k^i) \mid i \in \mathcal{V}, k = 0, 1, \ldots \} \), cannot be exactly inferred.

**Proof:** Assume that the eavesdropper collects information from \( K \) iterations to infer the information of agents. Since only partial links \( \mathcal{V} \) are observed, the measurement can be formulated as (19) but with extra unknown variables if the links in the Hamiltonian cycle are included in \( \mathcal{V} \). Hence from the proof of Theorem 2, we obtain a similar proof.

From the above analysis, we can conclude that the proposed PI-ADMM1 is privacy-preserving for the local functions of agents against the external eavesdropper. In addition, the privacy for agent \( i \in \mathcal{V} \) can also be guaranteed against other colluding agents.

**Theorem 3:** In PI-ADMM1, the primal and dual variables, and gradients of agent \( i \in \mathcal{V} \), i.e., \( \{ x_k^i, y_k^i, \nabla f_i(x_k^i) \mid k = 0, 1, \ldots \} \) cannot be inferred by any other honest-but-curious colluding agents in \( \mathcal{V} \). i.

**Proof:** Without loss of generality, we consider the privacy for agent \( i_0 \) against all the \( N - 1 \) colluding agents in \( \mathcal{V} \), which have knowledge of \( \{ x_k^i, y_k^i, z_k^i \mid i \in \mathcal{V}, k = 0, 1, \ldots \} \). Recalling the measurement (19), useful information for deducing the primal and dual variables, and gradients is given by

\[
\begin{aligned}
\begin{cases}
\begin{aligned}
\dot{x}_0^i &= y_0^i \rho \\
\dot{y}_0^i &= 0 \\
\dot{x}_{1+n}^i &= \frac{1}{1+\gamma_0^i} (N\Delta_{1+n}^i + \gamma_0^i N_{n_0} + x_{0}^{i+n}) \\
\dot{y}_{1+n}^i &= y_{n_0}^i + \frac{\rho\gamma_0^i}{1+\gamma_0^i} (z_{0}^{n} - N\Delta_{1+n}^i - x_{0}^{i+n})
\end{aligned}
\end{cases}
\end{aligned}
\]

(20)

where \( n \) is from 0 to \( \lfloor \frac{K}{2} \rfloor \). Since (20) contains \( 2\lfloor \frac{K}{2} \rfloor + 3 \) equations and \( 3\lfloor \frac{K}{2} \rfloor + 5 \) unknown variables, the remaining \( N - 1 \) agents cannot obtain the exact values of \( \{ x_k^i, y_k^i, \nabla f_i(x_k^i) \mid k = 0, \ldots, K \} \) by solving this under-determined system. With \( K \to \infty \), the KKT conditions reduce to

\[
\begin{aligned}
\nabla f_i(x_0^{K+1}) &= y_0^{K+1} \\
y_0^{K+1} &= -\sum_{j \in \mathcal{V}} y_j^{K+1} \\
z^{K+1} &= x_0^{K+1} + \sum_{j \in \mathcal{V}} y_j^{K+1}
\end{aligned}
\]

(21)

However, even combining (20) and (21), the exact values of \( \{ x_k^i, y_k^i, \nabla f_i(x_k^i) \mid k = 0, 1, \ldots \} \) cannot be obtained asymptotically. Thus, we have shown that the privacy of agent \( i_0 \) is guaranteed in the extreme case where all the agents in \( \mathcal{V} \) collude to infer the information of \( i_0 \). Hence we can claim that PI-ADMM1 is privacy preserving for agent \( i_0 \) against any other honest-but-curious colluding agents in \( \mathcal{V} \).

Since only the token is transmitted among agents in PI-ADMM1, the local variables and gradients \( \{ x_k^i, y_k^i, \nabla f_i(x_k^i) \} \) of agent \( i_0 \) cannot be observed by other agents directly. In fact, except \( \{ x_j^i, y_j^i, \nabla f_j(x_j^i) \mid j \in \mathcal{V} \} \), the information gathered by the eavesdropper is the same as that of colluding agents in \( \mathcal{V} \). This explains why the system (19) reduces to (20) in Theorem 3.

**V. NUMERICAL EXPERIMENTS**

In this section, we will conduct numerical experiments to evaluate the convergence and privacy properties of proposed PI-ADMM algorithms. For the simulation, we generate the connected network \( G \) with \( N \) agents and \( |E| = \frac{N(N-1)}{2} \) links. This ensures a Hamiltonian cycle in \( G \). We consider unicast among agents, and the resultant communication cost for each transmission of a \( p \)-dimensional vector is 1 unit.

We evaluate the convergence of proposed approaches with state-of-the-art approaches regarding the accuracy defined by

\[
\text{accuracy} = \frac{1}{N} \sum_{i=1}^{N} \frac{\| x_i^* - x_i \|^2}{\| x_i^* \|^2}
\]

(22)

where \( x_i^* \in \mathbb{R}^p \) is the optimal solution of (2). The dimension of \( x_i, y_i \) and \( z \) is set to be \( p = 2 \).

A. Decentralized Least-Squares Problem

We first consider the decentralized least-squares problem as [50], which aims at solving (1) with a local function

\[
\begin{aligned}
\min_{x;D_i} f_i(x_i; D_i) &= \frac{1}{2b_i} \sum_{j=1}^{b_i} \| x_i^T o_{i,j} - t_{i,j} \|^2
\end{aligned}
\]

(23)

where \( D_i = \{ o_{i,j}, t_{i,j} \mid j = 1, \ldots, b_i \} \) is the dataset of agent \( i \) locally. The entries of input \( o_{i,j} \in \mathbb{R}^2 \) and target \( t_{i,j} \in \mathbb{R}^2 \) follow i.i.d. distribution \( \mathcal{U}(0, 1) \). The number of data samples is kept unique across agents with \( b_i = 30 \).

We first simulate the convergence behavior of I-ADMM and W-ADMM for solving the decentralized least-squares problem in different graphs. From Fig. 3(a) in random graph with \( \eta = 0.2 \), I-ADMM is more communication efficient than W-ADMM. Even when \( G \) becomes complete graph (i.e., \( \eta = 1 \)), I-ADMM is still more efficient in terms of communication cost.
Fig. 3. The accuracy versus communication cost for I-ADMM and W-ADMM with $N = 100$ in: (a) random graphs; (b) ring graph.

Fig. 4. The accuracy of decentralized least-squares problem for W-ADMM ($\beta = 10$), IPW-ADMM ($\rho = 10, \tau = 0, M = 25$), COCA ($c = 1, \alpha = 1, \rho = 0.85$), I-ADMM ($\rho = 10$), PI-ADMM1 ($\rho = 10$) and PI-ADMM2 ($\rho = 10, \sigma = 10^{-3}$) with different network settings: (a) $N = 100, \eta = 0.3$; (b) $N = 100, \eta = 0.5$; (c) $N = 200, \eta = 0.3$.

Fig. 5. The estimation and true value in decentralized least-squares problem with respect to $\{x_1^k(1), y_1^k(1) | k = 0, 1, ..., 2000\}$ for I-ADMM, PI-ADMM1 and PI-ADMM2.

The accuracy over communication cost is shown in Fig. 4. It is clear that I-ADMM is the most communication efficient compared with W-ADMM [17], IPW-ADMM [18] and COCA [20] for different network settings. Equivalently, it demonstrates that I-ADMM has a faster convergence speed than W-ADMM. This is because the artificial noise added on the step size is scaled by the primal residue $z_{k+1} - x_{k+1}$, which converges to 0 gradually. Since the additive noise in PI-ADMM2 does not scale with iterations. As shown in Fig. 4, this introduces error bounds for the accuracy, which is determined by $\sigma$. Comparing sub-figures (a) and (b), the convergence behaviors of I-ADMM, PI-ADMM1 and PI-ADMM2 do not depend on the connectivity $\eta$. This is because the convergence property of these proposed algorithms is only determined by the size of Hamiltonian cycle. This also explains that the convergence speed degrades with the expanding networks.

Then we evaluate the capability of privacy preservation for proposed PI-ADMM algorithms with $N = 100$ and $\eta = 0.3$. Without loss of generality, we only focus on the estimation over $\{x_1^k(1), y_1^k(1) | k = 0, 1, ..., 2000\}$ with observations $\{z^k | k = 0, 1, ..., 2001\}$ for proposed methods. Besides, as for I-ADMM, we assume that the eavesdropper uses the recursive equation (12) to deduce the primal and dual variable of agent 1 with initialization $\{x_i = 0, y_i = 0 | i \in \mathcal{V}\}$ and inferring $x_{2001}^1$ with $z_{2001}$. The results are shown in Fig. 6(a) and (b), where the estimation coincides with the true value. The figures clearly demonstrate that I-ADMM is not privacy preserving. While for PI-ADMM1, we reformulate the under-determined equations (19) into the form $A\nu = b$, where
I-ADMM \((\beta = 1)\), IPW-ADMM \((\rho = 1, \tau = 3, M = 25)\), COCA \((c = 1, \alpha = 1, \rho = 0.85)\), I-ADMM \((\rho = 1)\), PI-ADMM1 \((\rho = 1)\) and PI-ADMM2 \((\rho = 1, \sigma = 10^{-3})\) with different network settings: (a) \(N = 100, \eta = 0.3\); (b) \(N = 100, \eta = 0.5\); (c) \(N = 200, \eta = 0.3\).

A similar procedure for estimation can also be established for PI-ADMM2. The estimation results for PI-ADMM1 and PI-ADMM2 are presented in Fig. 6(c)-(f). From sub-figures (c) and (e), for both PI-ADMM1 and PI-ADMM2, the gaps between the true value and estimation for \(x^k_i(1)\) shrinkages to 0 as convergence happens. This is inevitable and also consistent with analysis in Remark 4. However, the estimation error for \(y^k_i(1)\) diverges with iterations. We can see that PI-ADMM1 and PI-ADMM2 can prevent information leakage against the eavesdropper.

B. Decentralized Logistic Regression

In the decentralized logistic regression, the local loss function of agent \(i\) is

\[
 f_i(x_i) = \frac{1}{2b_i} \sum_{j=1}^{b_i} \log \left(1 + \exp \left(-t_{i,j}x_i^T\alpha_{i,j}\right)\right),
\]

where \(t_{i,j} \in \{-1, 1\}\) and \(b_i = 30\). Each sample feature \(\alpha_{i,j}\) follows \(N(0, I)\). To generate \(t_{i,j}\), we first generate a random vector \(x \in \mathbb{R}^2 \sim N(0, I)\). Then for each sample, we generate \(v_{i,j}\) according to \(U(0, 1)\), and if \(v_{i,j} \leq (1 + \exp(-x^T\alpha_{i,j}))^{-1}\), we set \(t_{i,j}\) as 1, otherwise \(-1\). Since it is difficult to solve the optimization problem (2) with (25) in I-ADMM based methods, we alternatively use the updating process (9). Fairly, we also adopt the first-order approximation for algorithms IPW-ADMM, W-ADMM and COCA.

In Fig. 5, we present the accuracy over communication costs for different network settings. Apparently, compared to W-ADMM, IPW-ADMM and COCA, the proposed I-ADMM based algorithms are the most communication-efficient. The curves with different network setups present the similar trends as those of Fig. 3.

The results regarding to privacy preservation are shown in Fig. 6. For PI-ADMM1 and PI-ADMM2, we set \(\{\mathbb{P}_i = \mathcal{U}(1 - \frac{0.1}{\rho_i}, 1 + \frac{0.1}{\rho_i})|i \in \mathcal{V}\}\) and initialize \(\{x_i^0 \sim \mathcal{U}(0, 10)|i \in \mathcal{V}\}\). The estimation approaches for I-ADMM, PI-ADMM1 and PI-ADMM2 follow those of Fig. 4. In Fig. 6, the sub-figures (a) and (b) demonstrate that I-ADMM cannot guarantee privacy for agents. As for PI-ADMM1 and PI-ADMM2, the states \(x^k_i(1)\) can be revealed with iteration \(k\) increasing. However, the estimation for \(y^k_i(1)\) diverges from the ground truth. Thus, proposed PI-ADMM algorithms are privacy preserving.
APPENDIX A
PROOF OF LEMMA 3

From steps 2–7 of PI-ADMM2, it is easy to verify that
\[
\frac{1}{N} \sum_{i=1}^{N} (x_i^0 - \bar{x}_i^0) = 0.
\]
Then we prove that \( \mathcal{L}_\rho(x^k, y^k, z^k) \) is non-increasing with iteration \( k \). From the optimality condition of (4a) and update (14), we can derive
\[
\nabla f_{ik}(x_{ik}^{k+1}) = y_{ik}^k + \rho \gamma_{ik}^k (z^{k} - x_{ik}^{k+1}) = y_{ik}^{k+1}.
\]
After updating to \( x_{ik}^{k+1} \) and \( y_{ik}^{k+1} \) by (4c), (14), we have
\[
\mathcal{L}_\rho(x^k, y^k, z^k) = \mathcal{L}_\rho(x_{ik}^{k+1}, y_{ik}^{k+1}, z_{ik}^{k+1})
\]
\[
\quad = f_{ik}(x_{ik}^k) - f_{ik}(x_{ik}^{k+1}) + \langle y_{ik}^{k+1}, z_{ik}^{k} - x_{ik}^{k+1} \rangle
\]
\[
\quad - \langle y_{ik}^{k+1}, z_{ik}^{k} - x_{ik}^{k+1} \rangle + \langle \rho (z^{k} - x_{ik}^{k+1}), x_{ik}^{k+1} - x_{ik}^{k} \rangle_B
\]
\[
\quad + \frac{\rho}{2} \| x_{ik}^{k+1} - x_{ik}^{k+1} \|^2,
\]
where \( a \) holds since the cosine identity \( \| b + c \|^2 = \| a + c \|^2 - 2 \langle b, c \rangle \). According to Assumption 2, term \( A \) in (27) satisfies
\[
A \geq \langle \nabla f_{ik}(x_{ik}^{k+1}), x_{ik}^{k+1} - x_{ik}^{k+1} \rangle - \frac{L}{2} \| x_{ik}^{k+1} - x_{ik}^{k+1} \|^2
\]
\[
= \langle y_{ik}^{k+1}, x_{ik}^{k+1} - x_{ik}^{k+1} \rangle - \frac{L}{2} \| x_{ik}^{k+1} - x_{ik}^{k+1} \|^2.
\]
Then \( A + B \) can be lower bounded by
\[
A + B \geq \langle y_{ik}^{k+1} - y_{ik}^{k+1}, z^{k} - x_{ik}^{k+1} \rangle + \frac{1}{\gamma_{ik}^k}
\]
\[
\times \langle y_{ik}^{k+1} - y_{ik}^{k+1}, x_{ik}^{k+1} - x_{ik}^{k+1} \rangle - \frac{L}{2} \| x_{ik}^{k+1} - x_{ik}^{k+1} \|^2
\]
\[
\geq - \frac{1}{\gamma_{ik}^k} \left( \frac{1}{\rho} + \frac{1}{2} \right) \| y_{ik}^{k+1} - y_{ik}^{k} \|^2
\]
\[
- \left( \frac{1}{\gamma_{ik}^k} + \frac{L}{2} \right) \| x_{ik}^{k+1} - x_{ik}^{k} \|^2,
\]
where \( a \) is because (30) and \( b \) are from Young’s inequality.
\[
\rho \left( z^{k} - x_{ik}^{k+1} \right) = \frac{1}{\gamma_{ik}^k} \left( y_{ik}^{k+1} - y_{ik}^{k} \right).
\]
Through updating token \( z \) at iteration \( k \), the change of augmented Lagrangian is measured by
\[
\mathcal{L}_\rho(x_{ik}^{k+1}, y_{ik}^{k+1}, z_{ik}^{k+1}) - \mathcal{L}_\rho(x_{ik}^{k+1}, y_{ik}^{k+1}, z_{ik}^{k+1})
\]
\[
= \sum_{i=1}^{N} \left[ \langle y_{ik}^{k+1}, z^{k} - z_{ik}^{k+1} \rangle + \frac{\rho}{2} \| z^{k} - x_{ik}^{k+1} \|^2 - \| z_{ik}^{k+1} - x_{ik}^{k+1} \|^2 \right]
\]
\[
= \frac{\rho N}{2} \| z_{ik}^{k+1} - z_{ik}^{k+1} \|^2
\]
\[
+ \sum_{i=1}^{N} \rho \left( z_{ik}^{k+1} - z_{ik}^{k+1} + \frac{y_{ik}^{k+1}}{\rho}, z_{ik}^{k} - z_{ik}^{k} \right)
\]
\[
= \frac{\rho N}{2} \| z_{ik}^{k+1} - z_{ik}^{k+1} \|^2,
\]
where \( a \) holds because the truth \( z_{ik}^{k+1} = \frac{1}{N} \sum_{i=1}^{N} (x_{ik}^{k+1} - y_{ik}^{k+1} / \rho) \). By combining (27) and (31), we obtain
\[
\mathcal{L}_\rho(x_{ik}^{k+1}, y_{ik}^{k+1}, z_{ik}^{k+1}) = \left( \frac{\rho - L}{2} - \frac{1}{2 \gamma_{ik}^k} \right) \| x_{ik}^{k+1} - x_{ik}^{k+1} \|^2 + \frac{\rho N}{2} \| z_{ik}^{k+1} - z_{ik}^{k+1} \|^2
\]
\[
- \frac{1}{\gamma_{ik}^k} \left( \frac{1}{\rho} + \frac{1}{2} \right) \| y_{ik}^{k+1} - y_{ik}^{k} \|^2
\]
\[
\geq \frac{\rho N}{2} \left( 1 - \frac{\rho N + 2N}{\gamma_{ik}^k} \right) \| z_{ik}^{k+1} - z_{ik}^{k+1} \|^2
\]
\[
+ \left( \frac{\rho - L}{2} - \frac{1}{2 \gamma_{ik}^k} - \frac{\rho^2 + 2 \rho}{\gamma_{ik}^k} \right) \| x_{ik}^{k+1} - x_{ik}^{k} \|^2,
\]
where \( a \) is from \( y_{ik}^{k+1} - y_{ik}^{k} = \rho (x_{ik}^{k+1} - x_{ik}^{k}) - \rho N (z_{ik}^{k+1} - z_{ik}^{k}) \) and triangle inequality. Hence to ensure that \( \mathcal{L}_\rho(x_{ik}^{k+1}, y_{ik}^{k+1}, z_{ik}^{k+1}) \) is non-increasing with iterations, we must have \( \rho > L \) and \( \gamma_{ik}^k > \max \left( \frac{2 \rho^2 + \rho L}{2 (\rho + 2 N)}, 2 (\rho + 2 N) \right) \). This proves property 1.

To prove statement 2), we verify that \( \mathcal{L}_\rho(x_{ik}^{k+1}, y_{ik}^{k+1}, z_{ik}^{k+1}) \) is lower bounded.
\[
\mathcal{L}_\rho(x_{ik}^{k+1}, y_{ik}^{k+1}, z_{ik}^{k+1})
\]
\[
= \sum_{j=0}^{N-1} \left[ f_{ik-j}(x_{ik-j}^{k-j+1}) + \langle y_{ik-j}^{k-j+1}, z_{ik-j}^{k-j+1} - x_{ik-j}^{k-j+1} \rangle \right.
\]
\[
+ \frac{\rho}{2} \| z_{ik-j}^{k-j+1} - x_{ik-j}^{k-j+1} \|^2 \left. \right]
\]
\[
\geq \sum_{j=0}^{N-1} \left[ f_{ik-j}(z^{k}) + \frac{\rho - L}{2} \| z^{k} - x_{ik-j}^{k-j+1} \|^2 \right.
\]
\[
- \left( \nabla f_{ik-j}(z_{ik-j}^{k-j+1}) - y_{ik-j}^{k-j+1}, z_{ik-j}^{k-j+1} - z_{ik-j}^{k-j+1} \right) \right].
\]
\[
\begin{align*}
&\geq \min_x \left\{ \sum_{i=1}^{N} f_i(x) \right\} + \frac{\rho - L}{2} \sum_{j=0}^{N-1} \left\| z^{k+1} - x^{k} - x^{k-j+1} \right\|^2 \\
&\geq F(x^*) > -\infty,
\end{align*}
\]
where (a) is from L-Lipschitz differentiable \(f_i\) and (b) is from the Assumption 2 in which the optimal value \(F(x^*)\) is assumed to be lower bounded. Guaranteeing \(\rho > L\) can make (33) satisfied, which completes the proof for property 2. Then with the monotonicity statement 1), we conclude that \(L_p\) is convergent.

**APPENDIX B**

**PROOF OF THEOREM 2**

Note that the KKT point \((x^*, y^*, z^*)\) of problem (2) satisfies the following conditions

\[
\begin{align*}
\nabla f_i(x_i^*) - y_i^* &= 0, \forall i \in \mathcal{V}; \\
y_i^* &= 0; \\
z^* &= x_i^*, \forall i \in \mathcal{V}.
\end{align*}
\]

Since \((34)\) also implies that \(\sum_{i=1}^{N} \nabla f_i(x_i^*) = 0\), \(x^*\) is also a stationary point of problem (2). By summing (32) from all iterations 0 to \(k\), we obtain

\[
\begin{align*}
\sum_{i=0}^{k} \left[ \rho N \left( \frac{1}{2} - \frac{\rho N + 2N}{\gamma_{i_1}} \right) \right] \left\| z^{i+1} - z^i \right\|^2 + \left( \frac{\rho - L}{2} - \frac{1}{2\gamma_{i_1}} - \frac{\rho^2 + 2\rho}{\gamma_{i_1}} \right) \left\| x^{i+1} - x^i \right\|^2
\end{align*}
\]
\[
\leq L_p \left( x^{0}, y^{0}, z^{0} \right) - L_p \left( x^{k+1}, y^{k+1}, z^{k+1} \right)
\]
\[
= [L_p \left( x^{0}, y^{0}, z^{0} \right) - F(x^*)] - [L_p \left( x^{k+1}, y^{k+1}, z^{k+1} \right) - F(x^*)]
\]
\[
\leq L_p \left( x^{0}, y^{0}, z^{0} \right) - F(x^*) < +\infty,
\]
where (a) is from (33). Then with the conditions \(\rho > L\) and \(\gamma_{i_k} > \max \{ \frac{2\rho^2 + 4\rho + 1}{\rho - L}, 2(\rho + 2)N \}\), the left hand side (LHS) of inequality (35) is positive and increasing with \(k\). Since the RHS of (35) is finite, the iterates \((x^k, y^k, z^k)\) must satisfy

\[
\begin{align*}
\lim_{k \to \infty} \left\| z^{k+1} - z^k \right\| &= 0; \\
\lim_{k \to \infty} \left\| x^{k+1} - x^k \right\| &= 0, \forall i \in \mathcal{V}.
\end{align*}
\]

In addition, from the inequality of (29), the dual residue is given by \(\lim_{k \to \infty} \| y^{k+1} - y^k \| \leq \lim_{k \to \infty} \rho \| x^{k+1} - x^k \| + \rho N \| z^{k+1} - z^k \| = 0\). Since \(\| y^{k+1} - y^k \| \geq 0\), we can conclude that

\[
\lim_{k \to \infty} \left\| y^{k+1} - y^k \right\| = 0, \forall i \in \mathcal{V}.
\]

Now we utilize (36) and (37) to show that the limit point of \((x^k, y^k, z^k)\) is a KKT point of problem (2). For any \(i \in \mathcal{V}\), there always exists a \(j \in [0, N - 1]\) satisfying

\[
\left\| z^{k+1} - x^k - x^{k-j+1} \right\| = \left\| z^k - x^k - x^{k-j} \right\| \\
\leq \left\| z^{k+1} - z^k \right\| + \left\| z^k - x^k - x^{k-j+1} \right\| \\
= \left\| z^{k+1} - z^k \right\| + \frac{1}{\rho} \left\| y^{k+1} - y^k \right\|,
\]
where (a) is from (30). Then we can conclude

\[
\lim_{k \to \infty} \left\| x^{k+1} - x^k \right\| = 0, \forall i \in \mathcal{V}.
\]

By applying (4c) and (26), we obtain that

\[
\lim_{k \to \infty} \sum_{i=1}^{N} y_i^k = 0; \lim_{k \to \infty} \nabla f_i(x_i^k) - y_i^k = 0, \forall i \in \mathcal{V}.
\]

Equations (39) and (40) imply that \((x^k, y^k, z^k)\) asymptotically satisfy the KKT conditions in (34). Then we conclude that the iterates \((x^k, y^k, z^k)\) in PI-ADMM1 have limit points, which satisfy KKT conditions of original problem (2).

**APPENDIX C**

**PROOF OF REMARK 5**

According to (4a), the states of agent \(i_k\) satisfy

\[
x_{i_k}^{K-nN+1} = x_{i_k}^{K-nN+2} = \ldots = x_{i_k}^{K-(n-1)N}, 1 \leq n \leq \left\lfloor \frac{K}{N} \right\rfloor.
\]

With \(\{\gamma_{i_k}^k = 1|k = 0, 1, \ldots\}\) in PI-ADMM1, we derive the estimate based on recursive equation (12). For \(k \in [K-nN+1, K-(n-1)N]\), it has

\[
x_{i_k}^{k} = x_{i_k}^{K-(n-1)N}
\]
\[
= 2x_{i_k}^{K-(n-1)N+1} - N\Delta x_{i_k}^{K-(n-1)N+1} - z_{i_k}^{K-(n-1)N+1}
\]
\[
= 4x_{i_k}^{(n-2)N+1} - 2N\Delta x_{i_k}^{(n-2)N+1} - 2z_{i_k}^{(n-2)N+1}
\]
\[
= 2^{n}x_{i_k}^{K+1} - \sum_{j=1}^{n} \left( \frac{N\Delta}{2} \right) z_{i_k}^{(n-j)N+1} - 2^{n-j} z_{i_k}^{K-(n-j)N+1}.
\]

By substituting \(x_{i_k}^{K+1}\) with \(z^{K+1}\) in (42), the measurement \(z^{K+1}\) is obtained. With the condition \(\| x^{K+1} - x_{i_k}^{K+1} \| < \epsilon\), the LHS of (18) is obtained. Since the eavesdropper cannot deduce \(y_{i_k}^{K+1}\) directly, instead \(y_{i_k}^{K+1}\) can be derived recursively from \(y_{i_k}^{0}\), where

\[
y_{i_k}^{K-nN+1} = y_{i_k}^{K-nN} + \frac{\rho}{2} \left( K-nN - N\Delta x_{i_k}^{K-nN+1} - x_{i_k}^{K-nN} \right).
\]
Due to $y_{iK}^0 = p_{xK}^0$ and the bounds for measurement $x_{iK}$, the final result is concluded.

REFERENCES

[1] Q. Ling and Z. Tian, “Decentralized sparse signal recovery for compressive sleeping wireless sensor networks,” IEEE Trans. Signal Process., vol. 58, no. 7, pp. 3816–3827, Jul. 2010.

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

[3] C. Ravazzi, S. M. Fossou, and E. Maglí, “Distributed iterative thresholding for $l_0/l_1$-regularized linear inverse problems,” IEEE Trans. Inf. Theory, vol. 61, no. 4, pp. 2081–2100, Apr. 2015.

[4] J. B. Predd, S. R. Kulkarni, and H. V. Poor, “A collaborative training algorithm for distributed learning,” IEEE Trans. Inf. Theory, vol. 55, no. 4, pp. 1856–1871, Apr. 2009.

[5] P. A. Forero, A. Cano, and G. B. Giannakis, “Consensus-based distributed support vector machines,” J. Mach. Learn. Res., vol. 11, no. 55, pp. 1663–1707, May 2010.

[6] G. Mateos, J. A. Bazerque, and G. B. Giannakis, “Distributed sparse linear regression,” IEEE Trans. Signal Process., vol. 58, no. 10, pp. 5262–5276, Oct. 2010.

[7] J. Chen and A. H. Sayed, “Diffusion adaptation strategies for distributed optimization and learning over networks,” IEEE Trans. Signal Process., vol. 60, no. 8, pp. 4289–4305, Aug. 2012.

[8] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, “Consensus in ad-hoc WSNs with noisy links-Part I: Distributed estimation of deterministic signals,” IEEE Trans. Signal Process., vol. 56, no. 1, pp. 350–364, Jan. 2008.

[9] G. B. Giannakis, V. Kekatos, N. Gatsis, S. Kim, H. Zhu, and B. F. Wollenberg, “Monitoring and optimization for power grids: A signal processing perspective,” IEEE Signal Process. Mag., vol. 30, no. 5, pp. 107–128, Sep. 2013.

[10] H. J. Liu, W. Shi, and H. Zhu, “Distributed voltage control in distribution networks: Online and robust implementations,” IEEE Trans. Smart Grid, vol. 9, no. 6, pp. 6106–6117, Nov. 2018.

[11] X. Mao, Y. Gu, and W. Yin, “Walk proximal gradient: An energy-efficient algorithm for consensus optimization,” IEEE Trans. Autom. Control, vol. 64, no. 2, pp. 2048–2060, Apr. 2019.

[12] K. Yuan, Q. Ling, and W. Yin, “On the convergence of decentralized gradient descent,” SIAM J. Optim., vol. 26, no. 3, pp. 1835–1854, 2016.

[13] 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.

[14] J. F. C. Mota, J. M. F. Xavier, P. M. Q. Aguilar, and M. Puschel, “D-ADMM: A communication-efficient distributed algorithm for separable optimization,” IEEE Trans. Signal Process., vol. 61, no. 10, pp. 2718–2727, May 2013.

[15] T. Chang, M. Hong, W. Liao, and X. Wang, “Asynchronous distributed ADMM for large-scale optimization: Part I: Algorithm and convergence analysis,” IEEE Trans. Signal Process., vol. 64, no. 12, pp. 3118–3130, Jun. 2016.

[16] R. Zhang and J. Kwok, “Asynchronous distributed ADMM for consensus optimization,” in Proc. Int. Conf. Mach. Learn., 2014, pp. 1701–1709.

[17] X. Mao, K. Yuan, Y. Hu, Y. Gu, A. H. Sayed, and W. Yin, “Walkman: A communication-efficient random-walk algorithm for decentralized optimization,” IEEE Trans. Signal Process., vol. 68, pp. 2513–2528, 2020.

[18] Y. Ye, H. Chen, Z. Ma, and M. Xiao, “Decentralized consensus optimization based on parallel random walk,” IEEE Commun. Lett., vol. 24, no. 2, pp. 391–395, Feb. 2020.

[19] Q. Ling, Y. Liu, W. Shi, and Z. Tian, “Weighted ADMM for fast decentralized network optimization,” IEEE Trans. Signal Process., vol. 64, no. 22, pp. 5930–5942, Nov. 2016.
[46] F. Li, B. Luo, and P. Liu, “Secure information aggregation for smart grids using homomorphic encryption,” in Proc. 1st IEEE Int. Conf. Smart Grid Commun., 2010, pp. 327–332.

[47] F. Yan, S. Sundaram, S. V. N. Vishwanathan, and Y. Qi, “Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties,” IEEE Trans. Knowl. Data Eng., vol. 25, no. 11, pp. 2483–2493, Nov. 2013.

[48] Y. Lou, L. Yu, S. Wang, and P. Yi, “Privacy preservation in distributed subgradient optimization algorithms,” IEEE Trans. Cybern., vol. 48, no. 7, pp. 2154–2165, Jul. 2018.

[49] S. Han, W. K. Ng, L. Wan, and V. C. Lee, “Privacy-preserving gradient-descent methods,” IEEE Trans. Knowl. Data Eng., vol. 22, no. 6, pp. 884–899, Jun. 2010.

[50] C. Huang, L. Liu, C. Yuen, and S. Sun, “Iterative channel estimation using LSE and sparse message passing for MmWave MIMO systems,” IEEE Trans. Signal Process., vol. 67, no. 1, pp. 245–259, Jan. 2019.

Yu Ye (Student Member, IEEE) received the bachelor’s degree in communication engineering and the master’s degree in communication and information systems from the University of Electronic Science and Technology of China, Chengdu, China, in 2012 and 2015, respectively. He received the Ph.D. degree from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2020. His current research interests include wireless caching, machine learning, and distributed optimization.

Hao Chen (Student Member, IEEE) received the B.Sc. degree in communication engineering and the M.Sc. degree in electronic and communication engineering from Soochow University, in 2014 and 2017, respectively. He is currently working toward the Ph.D. degree with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden. His current research interests include distributed optimization, edge computing and machine learning.

Ming Xiao (Senior Member, IEEE) received the bachelor’s and master’s degrees in engineering from the University of Electronic Science and Technology of China, ChengDu, in 1997 and 2002, respectively. He received Ph.D. degree from the Chalmers University of Technology, Sweden, on November 2007. From 1997 to 1999, he worked as a Network and Software Engineer with ChinaTelecom. From 2000 to 2002, he also held a Position in the SiChuan communications administration. Since November 2007, he has been with the Department of Information Science and Engineering, School of Electrical Engineering and Computer Science, Royal Institute of Technology, Sweden, where he is currently an Associate Professor. He was an Editor for IEEE TRANSACTIONS ON COMMUNICATIONS (2012–2017), IEEE COMMUNICATIONS LETTERS (Senior Editor, since January 2015) and IEEE WIRELESS COMMUNICATIONS LETTERS (2012–2016), and has been an Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS since 2018. He was the lead Guest Editor for IEEE ISAC Special issue on “Millimeter Wave Communications for future mobile networks” in 2017. He has been an Area Editor for IEEE Open Journal of the Communication Society since 2019.

Mikael Skoglund (Fellow, IEEE) received the Ph.D. degree from the Chalmers University of Technology, Sweden, in 1997. In 1997, he joined the Royal Institute of Technology (KTH), Stockholm, Sweden, where he was appointed to the Chair in Communication Theory in 2003. At KTH, he heads the Division of Information Science and Engineering, and the Department of Intelligent Systems.

Dr. Skoglund has worked on problems in source-channel coding, coding and transmission for wireless communications, Shannon theory, information and control, and statistical signal processing. He has authored and coauthored some 160 journal and 380 conference articles.

During 2003–2008, he was an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS and during 2008–2012, he was on the editorial board for the IEEE TRANSACTIONS ON INFORMATION THEORY. He was on numerous technical program committees for IEEE sponsored conferences, he was a General Co-Chair for IEEE ITW 2019, and he will serve as TPC Co-Chair for IEEE ISIT 2022.

H. Vincent Poor (Fellow, IEEE) received the Ph.D. degree in EECS from Princeton University, in 1977. From 1977 until 1990, he was on the faculty of the University of Illinois at Urbana-Champaign. Since 1990, he has been on the faculty at Princeton, where he is the Michael Henry Strater University Professor of Electrical Engineering. From 2006 until 2016, he was the Dean of Princeton’s School of Engineering and Applied Science. He has also held Visiting Appointments with several other institutions, including most recently with Berkeley and Cambridge. His research interests include the areas of information theory, machine learning and network science, and their applications in wireless networks, energy systems and related fields. Among his publications in these areas is the forthcoming book Advanced Data Analytics for Power Systems (Cambridge University Press, 2021).

Dr. Poor is a member of the National Academy of Engineering and the National Academy of Sciences, and is a Foreign Member of the Chinese Academy of Sciences, the Royal Society and other national and international academies. He received the Technical Achievement and Society Awards of the IEEE Signal Processing Society in 2007 and 2011, respectively. Recent recognition of his work includes the 2017 IEEE Alexander Graham Bell Medal, the 2019 ASEE Benjamin Garver Lamme Award, a D.Sc. honoris causa from Syracuse University, awarded in 2017, and a D.Eng. honoris causa from the University of Waterloo, awarded in 2019.