Privacy-preserving Decentralized Optimization via Decomposition

Chunlei Zhang and Huan Gao and Yongqiang Wang

Abstract—This paper considers the problem of privacy-preservation in decentralized optimization, in which $N$ agents cooperatively minimize a global objective function that is the sum of $N$ local objective functions. We assume that each local objective function is private and only known to an individual agent. To cooperatively solve the problem, most existing decentralized optimization approaches require participating agents to exchange and disclose estimates to neighboring agents. However, this results in leakage of private information about local objective functions, which is undesirable when adversaries exist and try to steal information from participating agents. To address this issue, we propose a privacy-preserving decentralized optimization approach based on proximal Jacobian ADMM via function decomposition. Numerical simulations confirm the effectiveness of the proposed approach.

I. INTRODUCTION

We consider the problem of privacy-preservation in decentralized optimization where $N$ agents cooperatively minimize a global objective function of the following form:

$$\min_{\tilde{x}} \quad f(\tilde{x}) = \sum_{i=1}^{N} f_i(\tilde{x}),$$

where variable $\tilde{x} \in \mathbb{R}^n$ is common to all agents, function $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is a private local objective function of agent $i$. This problem has found wide applications in various domains, ranging from rendezvous in multi-agent systems [1], support vector machine [2] and classification [3] in machine learning, source localization in sensor networks [4], to data regression in statistics [5], [6].

To solve the optimization problem (1) in a decentralized manner, different algorithms were proposed in recent years, including the distributed (sub)gradient algorithm [7], augmented Lagrangian methods (ALM) [8], and the alternating direction method of multipliers (ADMM) as well as its variants [8]–[11]. Among existing approaches, ADMM has attracted tremendous attention due to its wide applications [9] and fast convergence rate in both primal and dual iterations [11]. ADMM yields a convergence rate of $O(1/k)$ when the local objective functions $f_i$ are convex and a Q-linear convergence rate when all the local objective functions are strongly convex [12]. In addition, a recent work shows that ADMM can achieve a Q-linear convergence rate even when the local objective functions are only convex (subject to the constraint that the global objective function is strongly convex) [13].

On the other hand, privacy has become one of the key concerns. For example, in source localization, participating agents may want to reach consensus on the source position without revealing their position information [14]. In the rendezvous problem, a group of individuals may want to meet at an agreed time and place [1] without leaking their initial locations [15]. In the business sector, independent companies may want to work together to complete a common business for mutual benefit but without sharing their private data [16]. In the agreement problem [17], a group of organizations may want to reach consensus on a subject without leaking their individual opinions to others [15].

One widely used approach to enabling privacy-preservation in decentralized optimization is differential privacy [18]–[21] which protects sensitive information by adding carefully-designed noise to exchanged states or objective functions. However, adding noise also compromises the accuracy of optimization results and causes a fundamental trade-off between privacy and accuracy [18]–[20]. In fact, approaches based on differential privacy may fail to converge to the accurate optimization result even without noise perturbation [20]. It is worth noting that there exists some differential-privacy-based optimization approaches which are able to converge to the accurate optimization result in the mean-square sense, e.g. [22], [23]. However, those results require the assistance of a third party such as a cloud [22], [23], and therefore cannot be applied to the completely decentralized setting where no third parties exist. Encryption-based approaches are also commonly used to enable privacy-preservation [24]–[26]. However, such approaches unavoidably bring about extra computational and communication burden for real-time optimization [27]. Another approach to enabling privacy preservation in linear multi-agent networks is observability-based design [28], [29], which protects agents' information from non-neighboring agents through properly designing the weights of the communication graph. However, this approach cannot protect the privacy of adversary's direct neighbors.

To enable privacy in decentralized optimization without incurring large communication/computational overhead or compromising algorithmic accuracy, we propose a novel privacy solution through function decomposition. In the optimization literature, privacy has been defined as preserving the confidentiality of agents’ states [22], (sub)gradients or objective functions [20], [30], [31]. In this paper, we define privacy as the non-disclosure of agents’ (sub)gradients. We protect agents’ (sub)gradients because in many decentralized optimization applications, subgradients contain sensitive information such as salary or medical record [26], [30].

Contributions: We proposed a privacy-preserving decen-
centralized optimization approach through function decomposition. In contrast to differential-privacy based approaches which use noise to cover sensitive information and are subject to a fundamental trade-off between privacy and accuracy, our approach can enable privacy preservation without sacrificing accuracy. Compared with encryption-based approaches which suffer from heavy computational and communication burden, our approach incurs little extra computational and communication overhead.

Organization: The rest of this paper is organized as follows: Sec. II introduces the attack model and presents the proximal Jacobian ADMM solution to (1). Then a completely decentralized privacy-preserving approach to problem (1) is proposed in Sec. III. Rigorous analysis of the guaranteed privacy and convergence is addressed in Sec. IV and Sec. V, respectively. Numerical simulation results are provided in Sec. VI to confirm the effectiveness and computational efficiency of the proposed approach. In the end, we draw conclusions in Sec. VII.

II. BACKGROUND

We first introduce the attack model considered in this paper. Then we present the proximal Jacobian ADMM algorithm for decentralized optimization.

A. Attack Model

We consider two types of adversaries in this paper: Honest-but-curious adversaries and External eavesdroppers. Honest-but-curious adversaries are agents who follow all protocol steps correctly but are curious and collect all intermediate and input/output data in an attempt to learn some information about other participating agents [32]. External eavesdroppers are adversaries who steal information through wiretapping all communication channels and intercepting messages exchanged between agents.

B. Proximal Jacobian ADMM

The decentralized problem (1) can be formulated as follows: each $f_i$ in (1) is private and only known to agent $i$, and all $N$ agents form a bidirectional connected network, which is denoted by a graph $G = (V, E)$. $V$ denotes the set of agents, $E$ denotes the set of communication links (undirected edges) between agents, and $|E|$ denotes the number of communication links (undirected edges) in $E$. If there exists a communication link between agents $i$ and $j$, we say that agent $i$ and agent $j$ are neighbors and the link is denoted as $e_{i,j} \in E$ if $i < j$ is true or $e_{j,i} \in E$ otherwise. Moreover, the set of all neighboring agents of $i$ is denoted as $\mathcal{N}_i$ and the number of agents in $\mathcal{N}_i$ is denoted as $D_i$. Then problem (1) can be rewritten as

$$\min_{x \in \mathbb{R}^n, i \in \{1, 2, \ldots, N\}} \sum_{i=1}^{N} f_i(x_i)$$

subject to $x_i = x_j, \ \forall e_{i,j} \in E$,

where $x_i$ is a copy of $\bar{x}$ and belongs to agent $i$. Using the following proximal Jacobian ADMM [33], an optimal solution to (1) can be achieved at each agent:

$$x_i^{k+1} = \arg\min_{x_i} f_i(x_i) + \frac{\gamma_i \rho}{2} \| x_i - x_i^k \|^2$$

$$+ \sum_{j \in \mathcal{N}_i} (\lambda_{i,j}^k (x_i - x_j^k) + \frac{\rho}{2} \| x_i - x_j^k \|^2)$$

$$\lambda_{i,j}^{k+1} = \lambda_{i,j}^k + \rho(x_i^{k+1} - x_j^{k+1}), \ \forall j \in \mathcal{N}_i.$$
and the associated augmented Lagrangian function is
\[
L^k_{\rho}(x, \lambda) = \sum_{i=1}^{N} (f_{i}^{\alpha k}(x_i^\alpha) + f_{i}^{\beta k}(x_i^\beta)) + \sum_{e_{i,j} \in E} (\lambda_{i,j}^{\alpha} (x_i^\alpha - x_j^\alpha) + \frac{\rho}{2} \| x_i^\alpha - x_j^\alpha \|^2) + \sum_{i \in V} (\lambda_{i,i}^{\alpha\beta} (x_i^\alpha - x_i^\beta) + \frac{\rho}{2} \| x_i^\alpha - x_i^\beta \|^2),
\]
(6)
where \(x = [x_1^\alpha, x_1^\beta, x_2^\alpha, x_2^\beta, \ldots, x_N^\alpha, x_N^\beta] \in \mathbb{R}^{2Nn}\) is the augmented state. \(\lambda_{i,j}^{\alpha}\) is the Lagrange multiplier corresponding to the constraint \(x_i^\alpha = x_j^\alpha\), \(\lambda_{i,i}^{\alpha\beta}\) is the Lagrange multiplier corresponding to the constraint \(x_i^\alpha = x_i^\beta\), and all \(\lambda_{i,j}^{\alpha}\) and \(\lambda_{i,i}^{\alpha\beta}\) are stacked into \(\lambda, \rho\) is the penalty parameter, which is a positive constant scalar. It is worth noting that each agent does not need to know the associated augmented Lagrangian function (i.e., other agents’ objective functions) to update its states \(x_i^\alpha\) and \(x_i^\beta\), as shown below in (7) and (8).

Based on Jacobian update, we can solve (5) by applying the following iterations for \(i = 1, 2, \ldots, N:\)
\[
\begin{align*}
x_i^{\alpha(k+1)} &= \arg\min_{x_i^\alpha} \frac{\gamma_i^0 \rho}{2} \| x_i^\alpha - x_i^{\alpha(k)} \|^2 + L_{\rho}^{k+1}(x_i^\alpha, x_1^\alpha, \ldots, x_i^\alpha, \ldots, x_N^\alpha, x_1^\beta, \ldots, x_N^\beta, \lambda^k) \iota, \\
x_i^{\beta(k+1)} &= \arg\min_{x_i^\beta} \frac{\gamma_i^0 \rho}{2} \| x_i^\beta - x_i^{\beta(k)} \|^2 + L_{\rho}^{k+1}(x_i^\alpha, x_1^\alpha, \ldots, x_i^\beta, \ldots, x_N^\alpha, x_1^\beta, \ldots, x_N^\beta, \lambda^k) \iota.
\end{align*}
\]
(7)
(8)
Similarly, both \(\lambda_{i,j}^{\alpha\beta}\) and \(\lambda_{i,i}^{\alpha\beta}\) are introduced for the constraint \(x_i^\alpha = x_i^\beta\) in (7)-(11) to unify the algorithm description.

Next we give in detail our privacy-preserving function-decomposition based algorithm.

Algorithm I

**Initial Setup:** For all \(i = 1, 2, \ldots, N\), agent \(i\) initializes \(x_i^{\alpha0}\) and \(x_i^{\beta0}\), and exchanges \(x_i^{\alpha0}\) with neighboring agents. Then agent \(i\) sets \(x_i^{\alpha0} = x_i^{\beta0}\), \(\lambda_{i,j}^{\alpha\beta0} = x_i^{\alpha0} - x_j^{\alpha0}\), \(\lambda_{i,i}^{\alpha0\beta0} = x_i^{\alpha0} - x_i^{\beta0}\), and \(\lambda_{i,i}^{\alpha0\beta0} = x_i^{\beta0} - x_i^{\alpha0}\).

**Input:** \(x_i^{\alpha(k+1)}, \lambda_{i,j}^{\alpha\beta(k)}, \lambda_{i,i}^{\alpha\beta(k)}, \lambda_{i,j}^{\beta(k)}, \lambda_{i,i}^{\beta(k)}\).

**Output:** \(x_i^{\alpha(k+1)}, \lambda_{i,j}^{\alpha\beta(k+1)}, \lambda_{i,i}^{\alpha\beta(k+1)}, \lambda_{i,j}^{\beta(k+1)}, \lambda_{i,i}^{\beta(k+1)}\).

1. For all \(i = 1, 2, \ldots, N\), agent \(i\) constructs \(f_i^{\alpha(k+1)}\) and \(f_i^{\beta(k+1)}\) under the constraint \(f_i^{\alpha(k+1)} + f_i^{\beta(k+1)}\),
2. For all \(i = 1, 2, \ldots, N\), agent \(i\) updates \(x_i^{\alpha(k+1)}\) and \(x_i^{\beta(k+1)}\) according to the update rules in (7) and (8) respectively;
3. For all \(i = 1, 2, \ldots, N\), agent \(i\) sends \(x_i^{\alpha(k+1)}\) to neighboring agents;
4. For all \(i = 1, 2, \ldots, N\), agent \(i\) computes \(\lambda_{i,j}^{\alpha\beta(k+1)}\) and \(\lambda_{i,i}^{\alpha\beta(k+1)}\) according to (9)-(11);
5. Set \(k\) to \(k+1\), and go to 1.

**IV. PRIVACY ANALYSIS**

In this section, we rigorously prove that each agent’s gradient of local objective function \(\nabla f_j\) cannot be inferred by honest-but-curious adversaries and external eavesdroppers.

**Theorem 1:** In Algorithm I, agent’s gradient of local objective function \(\nabla f_j\) at any point except the optimal solution will not be revealed to an honest-but-curious agent.

**Proof:** Suppose that an honest-but-curious adversary agent \(i\) collects information from \(K\) iterations to infer the gradient \(\nabla f_j\) of a neighboring agent \(j\). The adversary agent \(i\) can establish \(2nK\) equations relevant to \(\nabla f_j\) by making use of the fact that the update rules of (7) and (8) are publicly known, i.e.,
\[
\begin{align*}
\nabla f_i^{\alpha0}(x_i^{\alpha0}) + (\gamma_i + D_j + 1) \rho x_i^{\alpha0} - \gamma_i \rho x_i^{\alpha0} + \sum_{m \in N_i} (\lambda_{j,m}^{\alpha0} - \rho x_m^{\alpha0}) + \lambda_{j,j}^{\beta0} - \rho x_j^{\beta0} &= 0, \\
\nabla f_i^{\beta1}(x_i^{\beta1}) + (\gamma_j + 1) \rho x_i^{\beta1} - \gamma_j \rho x_j^{\beta0} + \lambda_{j,j}^{\beta0} - \rho x_j^{\beta0} &= 0, \\
\vdots \\
\nabla f_i^{nK}(x_i^{\alphaK}) + (\gamma_j + D_j + 1) \rho x_i^{\alphaK} - \gamma_j \rho x_j^{\alphaK} + \sum_{m \in N_i} (\lambda_{j,m}^{\alphaK} - \rho x_m^{\alphaK}) + \lambda_{j,j}^{\betaK} - \rho x_j^{\betaK} &= 0,
\end{align*}
\]
(9)

In the system of \(2nK\) equations (12), \(\nabla f_i^{\alpha0}(x_i^{\alpha0}) (k = 1, 2, \ldots, K), \nabla f_i^{\beta1}(x_i^{\beta1}) (k = 1, 2, \ldots, K), \gamma_i, \gamma_j, \) and \(x_i^{\betak} (k = 0, 1, 2, \ldots, K)\) are unknown to adversary agent
i. Parameters $x_{mk}^i, m \neq j$ and $\lambda_{mk}^i, m \neq j$ are known to adversary agent $i$ only when agent $m$ and agent $i$ are neighbors. So the above system of $2nK$ equations contains at least $3nK + n + 2$ unknown variables, and adversary agent $i$ cannot infer the gradient of local objective function $\nabla f_j$ by solving (12).

It is worth noting that after the optimization algorithm converges, adversary agent $i$ can have another piece of information according to the KKT conditions [33]:

$$\nabla f_j(x_j^*) = - \sum_{m \in N_j} \lambda_{jm}^* \nabla f_m(x_m).$$

(13)

If agent $j$’s neighbors are also neighbors to agent $i$, the exact gradient of $f_j$ at the optimal solution can be inferred by an honest-but-curious agent $i$. Therefore, agent $j$’s gradient of local objective function $\nabla f_j$ will not be revealed to an honest-but-curious agent $i$ at any point except the optimal solution.

**Corollary 1:** In Algorithm I, agent $j$’s gradient of local objective function $\nabla f_j$ at any point except the optimal solution will not be revealed to external eavesdroppers.

**Proof:** The proof can be obtained following a similar line of reasoning of Theorem 1. External eavesdroppers can also establish a system of $2nK$ equations (12) to infer agent $j$’s gradient $\nabla f_j$. However, the number of unknowns $\nabla f_j(x_j)$ ($k = 1, 2, \ldots, K$), are up to $3nK + n + 2$, and $x_j(x_j)$ ($k = 0, 1, 2, \ldots, K$) adds up to $3nK + n + 2$. Therefore, external eavesdroppers cannot infer the gradient of local objective function $\nabla f_j$ at any point except the optimal solution.

**Remark 1:** It is worth noting that if multiple adversary agents cooperate to infer the information of agent $j$, they can only establish a system of $2nK$ equations containing at least $3nK + n + 2$ unknown variables as well. Therefore, our algorithm can protect the privacy of agents against multiple honest-but-curious adversaries and external eavesdroppers.

**V. CONVERGENCE ANALYSIS**

In this section, we rigorously prove the convergence of Algorithm I under the following assumptions:

**Assumption 1:** Each local function $f_i: \mathbb{R}^n \rightarrow \mathbb{R}$ is strongly convex and continuously differentiable, i.e.,

$$\nabla f_i(x) - \nabla f_i(y)^T(x - y) \geq m_i \|x - y\|^2.$$

In addition, there exists a lower bound $m_f > 0$ such that $m_i \geq 2m_f, \forall i = \{1, 2, \ldots, N\}$ is true.

**Assumption 2:** Each private local function $f_i: \mathbb{R}^n \rightarrow \mathbb{R}$ has Lipschitz continuous gradients, i.e.,

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L_f \|x - y\|.$$

**Assumption 3:** $f_i^{ak}$ is chosen under the following conditions:

1) $f_i^{ak}$ is strongly convex and differentiable, i.e.,

$$\nabla f_i^{ak}(x) - \nabla f_i^{ak}(y)^T(x - y) \geq m_f \|x - y\|^2.$$

2) $f_i^{ak} = f_i - f_i^{ak}$ is strongly convex and differentiable, i.e.,

$$\nabla f_i^{ak}(x) - \nabla f_i^{ak}(y)^T(x - y) \geq m_f \|x - y\|^2.$$

3) $f_i^{ak}$ has Lipschitz continuous gradients, i.e. there exists an $L < +\infty$ such that

$$\|\nabla f_i^{ak}(x) - \nabla f_i^{ak}(y)\| \leq L \|x - y\|.$$

4) $f_i^{ak} = f_i - f_i^{ak}$ has Lipschitz continuous gradients, i.e. there exists an $L < +\infty$ such that

$$\|\nabla f_i^{ak}(x) - \nabla f_i^{ak}(y)\| \leq L \|x - y\|.$$

5) $\lim_{k \to \infty} f_i^{ak} \to f_i^{ao}$ and $f_i^{ak}$ is bounded when $\hat{x}$ is bounded.

It is worth noting that under Assumption 1 and Assumption 2, $f_i^{ak}$ can be easily designed to meet Assumption 3. A quick example is $f_i^{ak}(x) = \frac{m_f}{2}x^T\hat{x} + b_i^T\hat{x}$ where $b_i \in \mathbb{R}^n$ can be time-varying, and satisfies $\lim_{k \to \infty} b_i^k \to b_i^*$ and $-\infty < \|b_i^k\| < +\infty$.

Because the function decomposition process amounts to converting the original network to a virtual network $G' = (V', E')$ of $2N$ agents, we relabel the local objective functions $f_i^{ak}$ and $f_i^{bk}$ for all $i = 1, 2, \ldots, N$ as $h_1^i, h_2^i, \ldots, h_{2N}^i$. We relabel the associated states $x_1^i, x_2^i, \ldots, x_{2N}^i$ for all $i = 1, 2, \ldots, N$. In addition, we relabel parameters $\gamma_1^i, \gamma_2^i, \ldots, \gamma_{2N}^i$. Then problem (4) can be rewritten as

$$\min_{x_i \in \mathbb{R}^n, i \in \{1, 2, \ldots, 2N\}} \sum_{i=1}^{2N} h_i^T(x_i)$$

subject to

$$Ax = 0$$

(14)

where $x = [x_1^T, x_2^T, \ldots, x_{2N}^T]^T \in \mathbb{R}^{2Nn}$ and $A = [a_{m,l}] \otimes I_n \in \mathbb{R}^{|E'| \times 2Nn}$ is the edge-node incidence matrix of graph.
$G'$ as defined in [34]. Parameter $a_{m,l}$ is defined as

$$a_{m,l} = \begin{cases} 1 & \text{if the } m^{th} \text{ edge originates from agent } l, \\ -1 & \text{if the } m^{th} \text{ edge terminates at agent } l, \\ 0 & \text{otherwise.} \end{cases}$$

We define each edge $e_{i,j}$ originating from $i$ and terminating at $j$ and denote an edge as $e_{i,j} \in E'$ if $i < j$ is true or $e_{j,i} \in E'$ otherwise.

Denote the iterating results in the $k$th step in Algorithm I as follows:

$$x^k = [x_1^{kT}, x_2^{kT}, \ldots, x_{2N}^{kT}]^T \in \mathbb{R}^{2Nn},$$

$$\lambda^k = [\lambda_{i,j}^k]_{i,j \in E'} \in \mathbb{R}^{|E'|n},$$

$$y^k = [x^{kT}, \lambda^{kT}]^T \in \mathbb{R}^{|E'|+2Nn}.$$ Further augment the coefficients $\gamma_i$ ($i = 1, 2, \ldots, 2N$) into the matrix form

$$U = \text{diag}(\gamma_1, \gamma_2, \ldots, \gamma_{2N}) \otimes I_n \in \mathbb{R}^{2Nn \times 2Nn},$$

and $D_i$ into the matrix form

$$D = \text{diag}(D_1, D_2, \ldots, D_{2N}) \otimes I_n \in \mathbb{R}^{2Nn \times 2Nn}.$$ Now we are in position to give the main results for this section:

**Lemma 1:** Let $x^*$ be the optimal solution, $\lambda^{k*}$ be the optimal multiplier to (14) at iteration $k$, and $y^{k*} = [x^{kT}, \lambda^{kT}]^T$. Further define $Q = U + D - A^T A$, $H = \text{diag}(\rho Q, \frac{1}{\rho} I_{|E'|n})$, and let $u > 1$ be an arbitrary constant, then we have

$$\parallel y^{k+1} - y^{k+1*} \parallel_H \leq \frac{\parallel y^k - y^{k+1*} \parallel_H}{\sqrt{1+\delta}}$$

if $U + D - A^T A$ is positive semi-definite and Assumptions 1, 2, and 3 are satisfied. In (15), $\parallel \hat{x} \parallel_H = \sqrt{\hat{x}^T H \hat{x}}$ and

$$\delta = \min\{\frac{(u-1)A_{\min}}{uQ_{\max}}, \frac{2n^2 A_{\min} \rho}{uL^2 + \rho^2 A_{\min} Q_{\max}}\},$$

where $Q_{\max}$ is the largest eigenvalue of $Q$, $A_{\min}$ is the smallest nonzero eigenvalue of $A^T A$, $m_r$ is the strongly convexity modulus, and $L$ is the Lipschitz modulus.

**Proof:** The results can be obtained following a similar line of reasoning in [35]. The detailed proof is given in the supplementary materials and can be found online [36].

**Lemma 3:** Let $x^*$ be the optimal solution, $\lambda^{k*}$ be the optimal multiplier to (14) at iteration $k$, and $y^{k*} = [x^{kT}, \lambda^{kT}]^T$. Further define $Q = U + D - A^T A$ and $H = \text{diag}(\rho Q, \frac{1}{\rho} I_{|E'|n})$, then we have

$$\| y^{k+1} - y^{k+1*} \|_H \leq \frac{\| y^k - y^{k+1*} \|_H}{\sqrt{1+\delta}} + p(k)$$

if $U + D - A^T A$ is positive semi-definite and Assumptions 1, 2, and 3 are satisfied.

**Proof:** The proof is provided in the Appendix.

**Remark 2:** It is worth noting that problem (14) is a reformulation of problem (1). So Theorem 2 guarantees that each agent’s state will converge to the optimal solution to (1).

VI. NUMERICAL EXPERIMENTS

We first present a numerical example to illustrate the efficiency of the proposed approach. Then we compare our approach with the differential-privacy based algorithm in [18]. We conducted numerical experiments on the following optimization problem.

$$\min_{\hat{x}} \sum_{i=1}^{N} \| \hat{x} - y_i \|^2$$

with $y_i \in \mathbb{R}^n$. Each agent $i$ deals with a private local objective function

$$f_i(x_i) = \| x_i - y_i \|^2, \forall i \in \{1, 2, \ldots, N\}.$$ We used the above optimization problem because it is easy to verify whether the obtained value is the optimal solution, which should be $\frac{\| y_{i=1}^{N} \|^2}{\sqrt{N}}$. Furthermore, (20) makes it easy to compare with [18], whose simulation is also based on (20).

A. Evaluation of Our Approach

To solve the optimization problem (20), $f_i^{b_1}(\hat{x})$ was set to $f_i^{b_1}(\hat{x}) = \frac{1}{2} \hat{x}^T \hat{x} + (b_i^1)^T \hat{x}$ for our approach in the simulations, where $b_i^1$ was set to $b_i^1 = \frac{1}{2}(c_i + d_i)$ with $c_i \in \mathbb{R}^n$ and $d_i \in \mathbb{R}^n$ being constants private to agent $i$. Fig. 4 visualizes the evolution of $x_i^{c_1}$ and $x_i^{d_2}$ ($i = 1, 2, \ldots, 6$) in one specific run where the network deployment is illustrated in Fig. 3. All $x_i^{c_1}$ and $x_i^{d_2}$ ($i = 1, 2, \ldots, 6$) converged to the optimal solution 13.758.
was computed as the average value of squared distances with respect to the optimal solution over $M$ runs [18], i.e.,

$$d = \frac{1}{6M} \sum_{i=1}^{6} \sum_{l=1}^{M} \| x_i^l - [0.35; 0.45] \|^2.$$  

Here $x_i^l$ is the obtained solution of agent $i$ in the $l$th run. For our approach, $x_i^l$ was calculated as the average of $x_i^{ol}$ and $x_i^{pl}$.

Simulation results from 5,000 runs showed that our approach converged to $[0.35; 0.45]$ with an error $d = 5.1 \times 10^{-4}$, which is negligible compared with the simulation results under the algorithm in [18] (cf. Fig. 5 where each differential privacy level was implemented for 5,000 times). The results confirm the trade-off between privacy and accuracy in differential-privacy based approaches.

**B. Comparison with the algorithm in [18]**

Under the network deployment in Fig. 3 we compared our privacy-preserving approach with the differential-privacy based algorithm in [18]. We simulated the algorithm in [18] under seven different privacy levels:

$$\epsilon = 0.2, 1, 10, 20, 30, 50, 100.$$  

In the objective function (20), $y_i$ was set to $y_i = [0.1 \times (i - 1) + 0.1 \times (i - 1) + 0.2]$. The domain of optimization for the algorithm in [18] was set to $\mathcal{X} = \{(x, y) \in \mathbb{R}^2 | x^2 + y^2 \leq 1 \}$. Note that the optimal solution $[0.35; 0.45]$ resided in $\mathcal{X}$. Detailed parameter settings for the algorithm in [18] were given as $n = 2, c = 0.5, q = 0.8, p = 0.9$, and

$$a_{ij} = \begin{cases} 0.2 & j \in \mathcal{N}_i, \\ 0 & j \notin \mathcal{N}_i, j \neq i, \\ 1 - \sum_{j \in \mathcal{N}_i} a_{ij} & i = j, \end{cases}$$ \hspace{1cm} (22)  

for $i = 1, 2, ..., 6$. In addition, the performance index $d$ in [18] was used to quantify the optimization error here, which was computed as the average value of squared distances with respect to the optimal solution over $M$ runs [18], i.e.,

$$\sqrt{1 + \delta^k} \cdot \| y^k - y^{k^*} \|_H \leq \| y^0 - y^{0^*} \|_H + \sum_{s=0}^{k-1} \sqrt{1 + \delta^s} p(s).$$  

(23)
Dividing both sides by $\sqrt{1+\delta^k}$, we have
\[
\|y^k - y^k_*\|_H \leq \frac{\|y^0 - y^0_*\|_H}{\sqrt{1+\delta^k}} + \sum_{s=0}^{k-1} \frac{1}{\sqrt{1+\delta^s}} p(s).
\]
(24)

It is clear $\lim_{k \to \infty} \frac{\|y^0 - y^0_*\|_H}{\sqrt{1+\delta^k}} = 0$ due to $\delta > 0$. Now our main goal is to prove $\lim_{k \to \infty} \frac{\sum_{s=0}^{k-1} \frac{1}{\sqrt{1+\delta^s}} p(s)}{} = 0$. Recall that in Assumption 3 we have $\lim_{k \to \infty} f_{i,k}^* \to f_{i,k}^{**}$. Therefore, we have $\lim_{k \to \infty} h_i^{k} \to h_i^{*}$ for all $i = 1, 2, \ldots, 2N$, i.e., $h_i^{k}$ converges to a fixed function $h_i^{*}$ ($h_i^{k}$ converges to a fixed function $h_i^{*}$).

On the other hand, we have
\[
p(k) = \frac{1}{\sqrt{pA^{\min}}} \|\nabla h^{k+1}(x^*) - \nabla h^k(x^*)\|.
\]
(25)

As a result of the convergence of $h_k$, we have $\lim_{k \to \infty} p(k) = 0$. Therefore, we have that $p(k) = \frac{1}{\sqrt{pA^{\min}}}$. In addition, we have
\[
\forall \varepsilon > 0, \quad \exists N_1 \in \mathbb{N}^+, \quad \text{s.t.} \quad |p(k)| \leq \varepsilon_1, \quad \forall k \geq N_1,
\]
where $\mathbb{N}^+$ is the set of positive integers. Further letting $\eta = \frac{1}{\sqrt{1+\delta}}$ and $F(k) = \sum_{s=0}^{k-1} \frac{1}{\sqrt{1+\delta^s}} p(s)$, we have $\eta \in (0, 1)$ and
\[
F(k) = \sum_{s=0}^{k-1} \eta^{k-s} p(s) = \sum_{s=0}^{N_1} \eta^{k-s} p(s) + \sum_{s=N_1+1}^{k-1} \eta^{k-s} p(s) \leq B \sum_{s=0}^{N_1} \eta^{k-s} + \varepsilon_1 \sum_{s=N_1+1}^{k-1} \eta^{k-s}
\]
\[
= B \eta^k \frac{\eta^{-N_1} - \eta^{-1}}{1 - \eta} + \varepsilon_1 \frac{\eta^{k-N_1-1} - \eta^{-1}}{1 - \eta} \leq B \eta^k \frac{\eta^{-N_1} - \eta}{1 - \eta} + \varepsilon_1 \frac{\eta^{-N_1-1} - \eta^{-1}}{1 - \eta}
\]
(26)

for $k \geq N_1 + 2$.

Recalling $\eta \in (0, 1)$, we have $\lim_{k \to \infty} B \eta^k \frac{\eta^{-N_1} - \eta}{1 - \eta} = 0$ and
\[
\forall \varepsilon = \varepsilon_1 > 0, \quad \exists N_2 \in \mathbb{N}^+, \quad \text{s.t.} \quad |B \eta^k \frac{\eta^{-N_1} - \eta}{1 - \eta}| \leq \varepsilon_1, \quad \forall k \geq N_2,
\]
(27)

Therefore, we can obtain
\[
\forall \varepsilon = \varepsilon_1 > 0, \quad \exists N = \max\{N_1, N_2\},
\]
\[
\text{s.t.} \quad |F(k)| \leq \varepsilon_1 + \varepsilon_1 \frac{\eta}{1 - \eta} = \frac{1}{1 - \eta} \varepsilon_1, \quad \forall k \geq N,
\]
(28)

which proves that $\lim_{k \to \infty} F(k) = 0$. Then according to (24), we have $\lim_{k \to \infty} \|y^k - y^k_*\|_H = 0$. Since $\|x^k - x^*\|_q \leq \|y^k - y^k_*\|_H$, we have $\lim_{k \to \infty} \|x^k - x^*\|_q = 0$ as well, which completes the proof.
[25] A. C. Yao. Protocols for secure computations. In 23rd Annual Symposium on Foundations of Computer Science, pages 160–164, 1982.

[26] C. L. Zhang, M. Ahmad, and Y. Wang. ADMM based privacy-preserving decentralized optimization. IEEE Transactions on Information Forensics and Security, 2018.

[27] R. L. Lagendijk, Z. Erkin, and M. Barni. Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation. IEEE Signal Processing Magazine, 30(1):82–105, 2013.

[28] S. Pequito, S. Kar, S. Sundaram, and A. P. Aguiar. Design of communication networks for distributed computation with privacy guarantees. In 2014 IEEE 53rd Annual Conference on Decision and Control, pages 1370–1376, 2014.

[29] A. Alaeddini, K. Morgansen, and M. Mesbahi. Adaptive communication networks with privacy guarantees. In American Control Conference, pages 4460–4465. IEEE, 2017.

[30] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. IEEE Transactions on Knowledge and Data Engineering, 25(11):2483–2493, 2013.

[31] Y. Lou, L. Yu, S. Wang, and P. Yi. Privacy preservation in distributed subgradient optimization algorithms. IEEE Transactions on Cybernetics, 2017.

[32] F. Li, B. Luo, and P. Liu. Secure information aggregation for smart grids using homomorphic encryption. In 2010 First IEEE International Conference on Smart Grid Communications, pages 327–332, 2010.

[33] W. Deng, M. Lai, Z. Peng, and W. Yin. Parallel multi-block ADMM with o(1/k) convergence. Journal of Scientific Computing, 71(2):712–736, 2017.

[34] E. Wei and A. Ozdaglar. Distributed alternating direction method of multipliers. In Proceedings of the 51st IEEE Conference on Decision and Control, pages 5445–5450, 2012.

[35] Q. Ling and A. Ribeiro. Decentralized dynamic optimization through the alternating direction method of multipliers. IEEE Transactions on Signal Processing, 62(5):1185–1197, 2014.

[36] C. L. Zhang, H. Gao, and Y. Wang. Privacy-preserving decentralized optimization via decomposition. arXiv preprint arXiv:1808.09566, 2018.
Supplemental Materials: Privacy-preserving Decentralized Optimization via Decomposition

B. Proof of Lemma 1

According to the update rules in (7) and (8), we have

\[ \nabla h_i^{k+1}(x_i^{k+1}) + \sum_{j \in N_i} (\lambda_{i,j}^k + \rho(x_{i,j}^{k+1} - x_j^k)) + \gamma_i \rho(x_i^{k+1} - x_i^*) = 0 \]  

(29)

for all \( i = 1, 2, \ldots, 2N \).

Let \((x^*, \lambda^{k*})\) be the Karush-Kuhn-Tucker (KKT) points for (14) at iteration \( k \), we have

\[ -A_i^T \lambda^{k*} = \nabla h_i^k(x_i^*) \]

\[ A_i x^* = 0 \]

(30)

where \( A_i \) indicates the columns of \( A \) corresponding to agent \( i \). It is worth noting that since \( \sum_{i=1}^{2N} h_i^k(x_i) \) is strongly convex, \( x^* \) is the optimal solution to (14).

Since each \( h_i^k \) is strongly convex, we have

\[ (\nabla h_i^{k+1}(x_i^{k+1}) - \nabla h_i^{k+1}(x_i^*))^T(x_i^{k+1} - x_i^*) \geq m_f \| x_i^{k+1} - x_i^* \|^2. \]  

(31)

Combining the above equation with (29) and (30), we have

\[ (-\sum_{j \in N_i} (\lambda_{i,j}^k + \rho(x_{i,j}^{k+1} - x_j^k)) - \gamma_i \rho(x_i^{k+1} - x_i^k) + A_i^T \lambda^{k+1*})^T(x_i^{k+1} - x_i^*) \geq m_f \| x_i^{k+1} - x_i^* \|^2. \]  

(32)

Noting that \( \lambda^{k+1}_i = \lambda_i^k + \rho(x_i^{k+1} - x_i^{k+1}) \), one has

\[ (-\sum_{j \in N_i} (\lambda_{i,j}^{k+1} + \rho(x_{i,j}^{k+1} - x_j^k)) - \gamma_i \rho(x_i^{k+1} - x_i^k) + A_i^T \lambda^{k+1*})^T(x_i^{k+1} - x_i^*) \geq m_f \| x_i^{k+1} - x_i^* \|^2. \]  

(33)

Based on the definition of \( A, D, \) and \( U \), one can further have

\[ \sum_{j \in N_i} \lambda_{i,j}^{k+1} = A_i^T \lambda^{k+1}, \]  

(34)

\[ \sum_{j \in N_i} (x_{i,j}^{k+1} - x_j^k) = (D - A_i^T A)_i^T(x^{k+1} - x^k), \]  

(35)

\[ \gamma_i \rho(x_i^{k+1} - x_i^k) = \rho U_i(x^{k+1} - x^k). \]  

(36)

Recall that \( Q = D - A_i^T A + U \) and \( Q = Q^T \), we can combine (33) with the above three equations to obtain

\[ (-A_i^T(\lambda^{k+1} - \lambda^{k+1*}) - \rho Q_i^T(x^{k+1} - x^k))^T(x_i^{k+1} - x_i^*) \geq m_f \| x_i^{k+1} - x_i^* \|^2. \]  

(37)

Summing both sides of (37) over \( i = 1, 2, \ldots, 2N \) and using

\[ \sum_{i=1}^{2N} (x_i^{k+1} - x_i^*)^T(A_i)(\lambda^{k+1} - \lambda^{k+1*}) = (x^{k+1} - x^*)^T A^T(\lambda^{k+1} - \lambda^{k+1*}) \]

\[ \sum_{i=1}^{2N} (x_i^{k+1} - x_i^*)^T \rho Q_i^T(x^{k+1} - x^k) = \rho(x^{k+1} - x^*)^T Q^T(x^{k+1} - x^k) \]

we have

\[ -(x^{k+1} - x^*)^T A^T(\lambda^{k+1} - \lambda^{k+1*}) - \rho(x^{k+1} - x^*)^T Q^T(x^{k+1} - x^k) \geq m_f \| x^{k+1} - x^* \|^2 \]  

(38)

Moreover, the following equalities can be obtained by using algebraic manipulations:

\[ (x^{k+1} - x^*)^T Q^T(x^{k+1} - x^k) = \frac{1}{2} \| x^{k+1} - x^k \|_Q^2 + \frac{1}{2} (\| x^{k+1} - x^* \|_Q^2 - \| x^k - x^* \|_Q^2), \]  

(39)

\[ (x^{k+1} - x^*)^T A^T(\lambda^{k+1} - \lambda^{k+1*}) = \frac{1}{\rho} (\lambda^{k+1} - \lambda^k)^T (\lambda^{k+1} - \lambda^{k+1*}) \]  

(40)

\[ \frac{1}{\rho} (\lambda^{k+1} - \lambda^k)^T (\lambda^{k+1} - \lambda^{k+1*}) = \frac{1}{2\rho} \| \lambda^{k+1} - \lambda^k \|^2 - \frac{1}{2\rho} \| \lambda^k - \lambda^{k+1*} \|^2 + \frac{1}{2\rho} \| \lambda^{k+1} - \lambda^{k+1*} \|^2 \]  

(41)
Based on the above three inequities, (38) can be rewritten as

\[
m_f \| x^{k+1} - x^* \|^2 \leq -\frac{\rho}{2} \| x^{k+1} - x^* \|^2 - \frac{1}{2\rho} \| x^k - x^* \|^2 + \frac{1}{2\rho} \| x^k - x^* \|^2 + \frac{1}{2\rho} \| x^k - x^* \|^2
\]

(42)

Recall that \( H = \text{diag}(\rho Q, \frac{1}{\rho} I_{[E|n]}) \) and \( y^k = [x^k Q, x^k]^{T} \). The above inequality can be simplified as

\[
2m_f \| x^{k+1} - x^* \|^2 \leq \| y^k - y^{k+1} \|^2_H - \| y^{k+1} - y^{k+1*} \|^2_H - \| y^{k+1} - y^k \|^2_H
\]

(43)

On the other hand, note that for any constant \( u > 1 \), the following relationship is true [35]

\[
(u - 1) \| a - b \|^2 \geq (1 - \frac{1}{u}) \| b \|^2 - \| a \|^2
\]

(44)

So we have

\[
(u - 1) \| \forall h^{k+1}(x^{k+1}) - \forall h^{k+1}(x^*) \|^2 = (u - 1) \| A^T(\lambda^{k+1} - \lambda^{k+1*}) + \rho Q^T(x^{k+1} - x^k) \|^2 \geq \frac{u - 1}{u} \| A^T(\lambda^{k+1} - \lambda^{k+1*}) \|^2 - \| \rho Q^T(x^{k+1} - x^k) \|^2
\]

(45)

Since \( \lambda^{k+1} \) and \( \lambda^{k+1*} \) lie in the column space of \( A \), we have [35]

\[
\| A^T(\lambda^{k+1} - \lambda^{k+1*}) \|^2 \geq A_{\text{min}} \| \lambda^{k+1} - \lambda^{k+1*} \|^2
\]

\[
\| \rho Q^T(x^{k+1} - x^k) \|^2 \leq \rho^2 Q_{\text{max}} \| x^{k+1} - x^k \|^2
\]

(46)

where \( Q_{\text{max}} \) is the largest eigenvalue of \( Q \), \( A_{\text{min}} \) is the smallest nonzero eigenvalue of \( A^T A \).

In addition, given that \( \| \forall h^{k+1}(x^{k+1}) - \forall h^{k+1}(x^*) \|^2 \leq L^2 \| x^{k+1} - x^* \|^2 \) true according to Assumption [3], using (46) and (45), we can obtain

\[
(u - 1) L^2 \| x^{k+1} - x^* \|^2 \geq (u - 1) A_{\text{min}} \| \lambda^{k+1} - \lambda^{k+1*} \|^2 - \rho^2 Q_{\text{max}} \| x^{k+1} - x^k \|^2
\]

(47)

Using algebraic manipulations, the above inequality can be rewritten as

\[
\frac{u Q_{\text{max}}}{(u - 1) A_{\text{min}}} \rho \| x^{k+1} - x^k \|^2 + \frac{u L^2}{\rho A_{\text{min}}} \| x^{k+1} - x^* \|^2 \geq \frac{1}{\rho} \| \lambda^{k+1} - \lambda^{k+1*} \|^2
\]

(48)

Adding \( \frac{u Q_{\text{max}}}{(u - 1) A_{\text{min}}} \rho \| \lambda^{k+1} - \lambda^k \|^2 \) and \( \rho Q_{\text{max}} \| x^{k+1} - x^* \|^2 \) to the left hand side of the above inequality, and adding \( \rho \| x^{k+1} - x^* \|^2 \) to the right hand side, we obtain the following inequality based on the fact \( \rho \| x^{k+1} - x^* \|^2 \leq \rho Q_{\text{max}} \| x^{k+1} - x^* \|^2 \):

\[
\frac{u Q_{\text{max}}}{(u - 1) A_{\text{min}}} \rho \| x^{k+1} - x^k \|^2 + \frac{1}{\rho} \| \lambda^{k+1} - \lambda^k \|^2 + \rho \| x^{k+1} - x^* \|^2 \geq \frac{1}{\rho} \| \lambda^{k+1} - \lambda^{k+1*} \|^2 + \rho \| x^{k+1} - x^* \|^2
\]

(49)

Let

\[
\delta = \min \{ \frac{(u - 1) A_{\text{min}}}{u Q_{\text{max}}}, \frac{2 m_f A_{\text{min}} \rho}{u L^2 + \rho^2 A_{\text{min}} Q_{\text{max}}} \}
\]

(50)

Inequality (49) becomes

\[
\frac{1}{\delta} \| y^{k+1} - y^k \|^2 + \frac{2 m_f}{\delta} \| x^{k+1} - x^* \|^2 \geq \| y^{k+1} - y^{k+1*} \|^2
\]

(51)

Based on (43) and (51), we can get

\[
\frac{1}{\delta} \| y^k - y^{k+1*} \|^2 \leq \frac{1}{\delta} \| y^{k+1} - y^{k+1*} \|^2 \geq \| y^{k+1} - y^{k+1*} \|^2
\]

(52)

which proves Lemma [1].
C. Proof of Lemma 2

First, we have

\[ \| y^k - y^{k+1*} \|_H - \| y^k - y^{k*} \|_H \leq \| y^{k+1*} - y^{k*} \|_H \]  \hspace{1cm} (53)

On the other hand, we have

\[ \| y^{k+1*} - y^{k*} \|_H = \frac{1}{\sqrt{\rho}} \| \lambda^{k+1*} - \lambda^{k*} \| \]  \hspace{1cm} (54)

\[ \| A^T (\lambda^{k+1*} - \lambda^{k*}) \| = \| \nabla h^{k+1}(x^*) - \nabla h^{k}(x^*) \| \]  \hspace{1cm} (55)

Therefore, we can get the following inequality using (46)

\[ \| \lambda^{k+1*} - \lambda^{k*} \| \leq \frac{1}{\sqrt{A_{\min}}} \| \nabla h^{k+1}(x^*) - \nabla h^{k}(x^*) \| \]  \hspace{1cm} (56)

Combing (53) to (56), we obtain

\[ \| y^k - y^{k+1*} \|_H \leq \| y^k - y^{k*} \|_H + \frac{1}{\sqrt{\rho A_{\min}}} \| \nabla h^{k+1}(x^*) - \nabla h^{k}(x^*) \| \]  \hspace{1cm} (57)

which completes the proof of Lemma 2.