Privacy-Preserving Distributed Projection LMS for Linear Multitask Networks

Chengcheng Wang, *Member, IEEE*, Wee Peng Tay, *Senior Member, IEEE*, Ye Wei, *Member, IEEE*, and Yuan Wang

***Abstract*—We develop a privacy-preserving distributed projection least mean squares (LMS) strategy over linear multitask networks, where agents' local parameters of interest or tasks are linearly related. Each agent is interested in not only improving its local inference performance via in-network cooperation with neighboring agents, but also protecting its own individual task against privacy leakage. In our proposed strategy, at each time instant, each agent sends a noisy estimate, which is its local intermediate estimate corrupted by a zero-mean additive noise, to its neighboring agents. We derive a sufficient condition to determine the amount of noise to add to each agent's intermediate estimate to achieve an optimal trade-off between the network mean-square-deviation and an inference privacy constraint. We propose a distributed and adaptive strategy to compute the additive noise powers, and study the mean and mean-square behaviors and privacy-preserving performance of the proposed strategy. Simulation results demonstrate that our strategy is able to balance the trade-off between estimation accuracy and privacy preservation.**

***Index Terms*—Distributed strategies, multitask networks, inference privacy, privacy preservation, additive noises.**

I. Introduction

In multitask distributed networks, a set of interconnected agents work collaboratively to estimate different but related parameters of interest [1]–[8]. In order to make use of the relationship between different tasks for better inference performance, local estimates are exchanged amongst agents within the same neighborhood. However, each agent may wish to protect its own local parameters of interest and prevent other agents in the network from accurately inferring these parameters. For example, in an Internet of Things (IoT) network, sensors are deployed in smart grids, traffic monitoring, health monitoring, home monitoring and other applications [9]–[12]. Although different IoT or edge computing devices may have their local objectives, they can exchange information with each other or service providers [13]–[16] to improve inferences and services. However, sharing information may lead to privacy leakage. In addition, privacy concerns also exist in the areas of collaborative machine learning [17], [18] and related techniques such as federated learning [19], [20], which allow multiple participants, each with his own training dataset, to build a joint model by training locally and periodically exchanging model updates [17]. Data at different participants may come from different distributions which results in multiple learning tasks [18], [20]. Recent works have demonstrated that communicating model updates throughout the training process can reveal sensitive information [17], [19], [21].

To protect the privacy of data being exchanged between agents in a distributed network, the works [22]–[27] propose local differential privacy mechanisms, while [28]–[31] develop privacy-preserving distributed data analytics. However, these approaches often lead to significant trade-offs in estimation accuracy as they do not specifically protect the privacy of the parameters of interest. Privacy-preserving average consensus algorithms in [32]–[36] aim to guarantee the privacy of initial states while achieving asymptotic consensus on the average of the initial values. To achieve inference privacy in a decentralized IoT network, [37]–[41] propose approaches with information privacy guarantees, while [42]–[48] map the agents' raw observations into a lower dimensional subspace. These inference privacy works assume that all agents in the network are interested in inferring the same parameters or hypothesis of interest.

Existing works on multitask distributed networks mainly focus on developing new distributed strategies for automatic clustering [2], [49] and handling different relationships among the tasks [1], [4]–[8]. Other works like [2], [50]–[52] focus on evaluating the performance of different distributed schemes. Few works have considered protecting the privacy of each agent's local parameters. The reference [53] considers data privacy of the agents' local measurements in a single-task network.

Our objective is to develop a privacy-preserving distributed projection least mean squares (LMS) strategy over multitask networks, which balances the trade-off between estimation accuracy and privacy preservation of agents' local parameters or tasks. Specifically, we consider multitask estimation problems where the unknown parameters of interest within each neighborhood are linearly related with each other [7]. Such problems exist widely in applications such as electrical networks, telecommunication networks, and pipeline networks. Different from the strategy in [7], which does not take privacy preservation into consideration, we propose to sanitize each agent's intermediate estimate before sharing it

This work was supported in part by the Singapore Ministry of Education Academic Research Fund Tier 2 grant MOE2018-T2-2-019 and by A*STAR under its RIE2020 Advanced Manufacturing and Engineering (AME) Industry Alignment Fund – Pre Positioning (IAF-PP) (Grant No. A19D6a0053).

C. Wang and W. P. Tay are with Nanyang Technological University, Singapore. Y. Wei is with Northeast Electric Power University, China. Y. Wang is with Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore. E-mails: wangcc@ntu.edu.sg; wptay@ntu.edu.sg; weiye@hrbeu.edu.cn; wang_yuan@ihpc.a-star.edu.sg.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes a list of the common symbols used in the text and proofs of several theoretical findings in this paper. This material is 261 KB in size.

with its neighbors by adding an appropriate zero-mean noise to the intermediate estimate. We study how to design the power of the noise to be added to optimize the trade-off between the network mean-square-deviation (MSD) and the inference privacy of each agent's local parameters, measured by its neighbors' mean-square error in estimating the agent's local parameters. We extend the preliminary results in our conference paper [54] from the following three aspects: 1) We develop techniques to mitigate the negative effect of additive noises on the estimation accuracy of agents' local parameters. 2) We motivate a distributed and adaptive scheme that enables each agent in the network to estimate the power of the additive noise locally by using data available to it at each time instant. 3) We study the mean and mean-square behaviors of the proposed algorithm and evaluate its privacy-preserving performance. In addition, different from existing works on distributed strategies in the presence of link noises [51], [52], [55], [56], which examine the effect of link noises on the performance of distributed strategies, this work focuses on developing agent-specific time-varying variances of the additive noises that enable agents to benefit from in-network cooperation as well as to protect their own individual tasks against privacy leakage.

The rest of this paper is organized as follows. In Section II, we introduce the system assumptions and the multitask estimation problem considered in this paper. In Section III, we propose a privacy-preserving multitask distributed strategy by introducing privacy noises and show how to obtain the noise powers in Section IV. We examine the mean and mean-square behaviors and privacy-preserving performance of the proposed strategy in Section V. We present simulation results in Section VI. Section VII concludes the paper.

Notations: We use lowercase letters to denote vectors and scalars, uppercase letters for matrices, plain letters for deterministic variables, and boldface letters for random variables. We use $\mathbb{R}$ to denote field of real numbers, and $\mathbb{R}^{M\times N}$ for an $M\times N$ real matrix. We use $(\cdot)^\intercal$, $(\cdot)^{-1}$, $\mathrm{Tr}(\cdot)$ and $\mathrm{rank}(\cdot)$ to denote matrix transpose, inversion, trace and rank, respectively. We use $\mathrm{diag}\{\cdot\}$, $\mathrm{col}\{\cdot\}$ and $\mathrm{row}\{\cdot\}$ for a diagonal matrix, column vector and row vector, respectively. The $M\times M$ identity matrix is denoted as I_M. The $N\times 1$ vector of ones is denoted as $\mathbf{1}_{N\times 1}$, and $\mathbf{0}_{M\times N}$ denotes an $M\times N$ zero matrix. We have $\|x\|_\Sigma^2 = x^\intercal\Sigma x$ with Σ being a symmetric positive semidefinite matrix, and let $\|x\|^2 = \|x\|_{I_M}^2$ where M is the length of x. In addition, we use $\rho(A)$ to denote the spectral radius of matrix A, and $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ for the minimum and maximum eigenvalues of the symmetric matrix A, respectively. For a block matrix A whose (k,ℓ)-th block represents some interaction between agents k and ℓ, we let $[A]_k$ and $[A]_{\cdot,k}$ denote the block row and block column of A corresponding to agent k, respectively, and $[A]_{k,\ell}$ be the (k,ℓ)-th block. A set $\{a_1,a_2,\ldots,a_N\}$ is often denoted as $\{a_k\}$ for convenience if its meaning is clear from the context. For ease of reference, we list the common symbols used in the text and supplementary material in Table S1 of the supplementary material.

II. Linear Multitask Networks

In this section, we present our system model, and give a brief introduction to multitask networks, where neighboring agents' tasks are linearly related. Consider a strongly-connected network of N agents, where information can flow in either direction between any two connected agents [51]. At each time instant $i\geq 0$, each agent k has access to a real-valued scalar observation $\boldsymbol{d}_k(i)$, and a real-valued $M_k\times 1$ regression vector $\boldsymbol{u}_k(i)$. The random data $\{\boldsymbol{d}_k(i),\boldsymbol{u}_k(i)\}$ are related via the linear regression model

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_k^\intercal(i)\boldsymbol{w}_k^o + \boldsymbol{v}_k(i) \tag{1}$$

where the scalar $\boldsymbol{v}_k(i)$ is measurement noise, $\boldsymbol{w}_k^o$ is an $M_k\times 1$ unknown random vector, with mean $\mathbb{E}\boldsymbol{w}_k^o$ and covariance matrix

$$W_{kk} = \mathbb{E}[(\boldsymbol{w}_k^o-\mathbb{E}\boldsymbol{w}_k^o)(\boldsymbol{w}_k^o-\mathbb{E}\boldsymbol{w}_k^o)^\intercal]\in\mathbb{R}^{M_k\times M_k}. \tag{2}$$

Although we assume that the parameter vector $\boldsymbol{w}_k^o$ is random instead of being a deterministic parameter vector, like most of the literature on distributed strategies [1], [2], [4]–[7], [49]–[53], [55]–[57], we assume that the parameter vector $\boldsymbol{w}_k^o$ is fixed at a certain realization w_k^o during the distributed estimation process. Since our goal is to develop inference privacy mechanisms that lead to high estimation errors, on average, of agent k's local parameters $\boldsymbol{w}_k^o$ by other agents $\ell\neq k$, we adopt a Bayesian framework for the privacy criterion [37]–[40].

We make the following assumptions regarding model (1).

Assumption 1. *(Measurement noise) The measurement noise $\boldsymbol{v}_k(i)$ is white over time, with zero mean, and a variance of $\sigma_{v,k}^2$.*

Assumption 2. *(Regressors) The regressors $\{\boldsymbol{u}_k(i)\}$ are zero-mean, white over time and space with*

$$\mathbb{E}\boldsymbol{u}_k(i)\boldsymbol{u}_\ell^\intercal(j) = R_{u,k}\delta_{k,\ell}\delta_{i,j}, \tag{3}$$

where $R_{u,k}$ is symmetric positive definite, and $\delta_{k,\ell}$ is the Kronecker delta.

Assumption 3. *(Independence) The random data $\{\boldsymbol{w}_k^o,\boldsymbol{u}_\ell(i),\boldsymbol{v}_m(j)\}$ are independent of each other for any agents k,ℓ,m and any time instants i,j.*

For a realization $\{\boldsymbol{w}_k^o = w_k^o : k=1,\ldots,N\}$, the objective of each agent k is to find the minimizer of the following mean-square-error cost function:

$$J_k(w_k) = \mathbb{E}\big[(\boldsymbol{d}_k(i)-\boldsymbol{u}_k^\intercal(i)w_k)^2\,\big|\,\boldsymbol{w}_k^o = w_k^o\big]. \tag{4}$$

Let $\mathcal{N}_k$ be the set of all neighbors of agent k, including agent k itself, and

$$\boldsymbol{w}_{\mathcal{N}_k}^o = \mathrm{col}\{\boldsymbol{w}_\ell^o\}_{\ell\in\mathcal{N}_k}.$$

Following [7], we assume that neighboring tasks $\{\boldsymbol{w}_\ell^o : \ell\in\mathcal{N}_k\}$ of any agent $k=1,\ldots,N$ are related via

$$\mathcal{D}_k\boldsymbol{w}_{\mathcal{N}_k}^o + b_k = \mathbf{0}_{j_k\times 1}, \tag{5}$$

where the matrix $\mathcal{D}_k$ is a $j_k\times|\mathcal{N}_k|$ block matrix and the vector b_k is a $j_k\times 1$ block vector with $j_k\geq 1$. We note that the linear

equality relationship (5) exists in flow networks (see Section VI-E for details of flow networks and formulation of the linear equality relationships amongst neighboring tasks). Such networks are applicable to transportation networks, electrical networks, telecommunication networks, and pipeline networks [7].

Note that the local linear equality constraint (5) includes all j_k linear equality constraints that agent k is involved in. We assume that each agent k in the network is involved in at least one constraint so that cooperation is justified. Let Q be the total number of linear equality constraints that agents in the network are involved in, and $\mathcal{I}_q$ denote the set of agents involved in the q-th constraint for any $q = 1, \ldots, Q$. Then, for any two agents $\{\ell, k\} \subset \mathcal{I}_q$, we have $\ell \in \mathcal{N}_k$, and $k \in \mathcal{N}_\ell$. This enables each agent $k \in \mathcal{I}_q$ to collect estimates from all agents in $\mathcal{I}_q$ in order to satisfy the q-th constraint, which allows a distributed processing of each agent's local constraints.

The objective for the entire network is to find the optimal solution to the following constrained optimization problem [7]:

$$\min_{w_1,\ldots,w_N} \sum_{k=1}^{N} J_k(w_k) \tag{6a}$$

$$\text{s.t.} \quad w_{\mathcal{N}_k} = \text{col}\{w_\ell\}_{\ell\in\mathcal{N}_k}, \tag{6b}$$

$$\mathcal{D}_k w_{\mathcal{N}_k} + b_k = \mathbf{0}_{j_k\times 1}, \text{ for } k = 1, \ldots, N, \tag{6c}$$

where individual costs $\{J_k(w_k)\}$ are defined by (4). We assume that the optimization problem (6) is feasible [7]. We assume that the matrix $\mathcal{D}_k$ for any agent k is full row-rank so that the linear system (6c) has at least one solution. To avoid having a trivial solution, we also assume that it is column-rank deficient because otherwise (6c) has a unique solution.

As demonstrated in [7], each agent k benefits through cooperation with neighboring agents by sharing its local parameter estimate $\boldsymbol{\psi}_k(i)$ with its neighbors at each time instant i. By leveraging the linear relationships (5) and its neighbors' parameter estimates, an agent can improve its own inference accuracy. In this paper, we consider the scenario where agent k also wants to prevent other agents from inferring its own task $\boldsymbol{w}_k^o$. Thus, a privacy-preserving distributed solution is required to balance the trade-off between estimation accuracy and privacy protection of the individual tasks. In order to limit privacy leakage, we add a zero-mean, independent and identically distributed (i.i.d.) noise vector $\boldsymbol{n}_k(i)$ to agent k's local parameter estimate $\boldsymbol{\psi}_k(i)$, which is a local linear estimate of $\boldsymbol{w}_k^o$ (cf. (9), (11) and (12)), before communicating $\boldsymbol{\psi}'_k(i) = \boldsymbol{\psi}_k(i) + \boldsymbol{n}_k(i)$ to neighboring agents. Let $\sigma^2_{n,k}(i)$ be the variance of each random entry in $\boldsymbol{n}_k(i)$. We call $\boldsymbol{n}_k(i)$ a *privacy noise*.

Our objective is to find the optimal solution to the following optimization problem:

$$\min_{\sigma^2_{n,1}(i),\ldots,\sigma^2_{n,N}(i)} \quad \text{MSD}_{\text{net}}(i) = \frac{1}{N}\sum_{k=1}^{N} \mathbb{E}\|\boldsymbol{w}_k^o - \boldsymbol{w}_k(i)\|^2, \tag{7a}$$

$$\text{s.t.} \quad \mathbb{E}\|\boldsymbol{w}_k^o - \widehat{\boldsymbol{w}}_{k|\boldsymbol{\psi}'_k}(i)\|^2 \geq \delta_k, \tag{7b}$$

for $k = 1, \ldots, N, i \geq 0$, and where the $M_k \times 1$ coefficient vector $\boldsymbol{w}_k(i)$ denotes an estimate of $\boldsymbol{w}_k^o$ at agent k and time instant i, and $\widehat{\boldsymbol{w}}_{k|\boldsymbol{\psi}'_k}(i)$ is the least mean-square estimate of $\boldsymbol{w}_k^o$ at time instant i based on $\boldsymbol{\psi}'_k(i)$, and $\delta_k \geq 0$ is a privacy threshold chosen according to privacy requirements. Note that the privacy thresholds have to be chosen such that

$$0 \leq \delta_k \leq \text{Tr}(W_{kk}), \tag{8}$$

otherwise (7) is infeasible.

Remark 1: In (7b), it is required that at each time instant $i \geq 0$, the *expected* squared distance $\mathbb{E}\|\boldsymbol{w}_k^o - \widehat{\boldsymbol{w}}_{k|\boldsymbol{\psi}'_k}(i)\|^2$ over all realizations of $\boldsymbol{w}_k^o$ is no smaller than the predefined parameter δ_k. This provides an inference privacy constraint on the ability of a neighboring agent to agent k in accurately estimating $\boldsymbol{w}_k^o$ based on observation $\boldsymbol{\psi}'_k(i)$ on average.

Remark 2: In (7b), the estimator $\widehat{\boldsymbol{w}}_{k|\boldsymbol{\psi}'_k}(i)$ is based only on the noisy estimate $\boldsymbol{\psi}'_k(i)$. For each neighboring agent $\ell \in \mathcal{N}_k$, it has access to not only the received noisy estimate $\boldsymbol{\psi}'_k(i)$ from agent k, but also its own intermediate estimate $\boldsymbol{\psi}_\ell(i)$ of $\boldsymbol{w}_\ell^o$. Both of these estimates can be used to infer $\boldsymbol{w}_k^o$ as the unknown parameters $\boldsymbol{w}_k^o$ and $\boldsymbol{w}_\ell^o$ are linearly related with each other through (6c). In this paper, to simplify the analysis and to ensure that the sequence of noise variances $\{\sigma^2_{n,k}(i)\}_{i\geq 0}$ at each agent k is bounded and convergent, we only consider the simplified case in (7). We illustrate the case where a neighboring agent uses its own estimates as additional information using simulations in Section VI.

III. Privacy-preserving Distributed Projection LMS

In this section, we propose an inference privacy mechanism to protect each agent's local task by adding noise to its intermediate estimate before sharing with its neighbors. We then introduce a *weighted* projection operator, which projects neighbors' noisy estimates onto the linear manifold defined by local constraints (5), in order to mitigate the negative effect of the additive noises on the estimation accuracy of individual tasks.

A. Adapt-Then-Project Strategy

In our privacy-preserving distributed projection LMS algorithm, we initialize $\boldsymbol{w}_k(-1) = \mathbf{0}_{M_k\times 1}$ for every agent k in the network. Given data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_k(i)\}$ for each time instant $i \geq 0$, and for each agent $k = 1, \ldots, N$, we perform the following steps iteratively:

1) Adaptation. Each agent k updates its current estimate $\boldsymbol{w}_k(i-1)$ using the stochastic gradient descent (SGD) algorithm to obtain an intermediate estimate

$$\boldsymbol{\psi}_k(i) = \boldsymbol{w}_k(i-1) + \mu_k \boldsymbol{u}_k(i)\left(\boldsymbol{d}_k(i) - \boldsymbol{u}_k^{\mathsf{T}}(i)\boldsymbol{w}_k(i-1)\right), \tag{9}$$

where $\boldsymbol{\psi}_k(i)$ is of dimension $M_k \times 1$ and $\mu_k > 0$ is the step-size at agent k.

2) Exchange. Each agent k sends a noisy estimate

$$\boldsymbol{\psi}'_k(i) = \boldsymbol{\psi}_k(i) + \boldsymbol{n}_k(i), \tag{10}$$

where the random additive noise vector $\boldsymbol{n}_k(i)$ is of dimension $M_k \times 1$, to its neighbors and collects estimates $\{\boldsymbol{\psi}'_\ell(i)\}$ from neighboring agents $\{\ell \in \mathcal{N}_k\}$:

$$\boldsymbol{\psi}'_\ell(i) = \begin{cases} \boldsymbol{\psi}_\ell(i) + \boldsymbol{n}_\ell(i), & \text{if } \ell \in \mathcal{N}_k \backslash \{k\}, \\ \boldsymbol{\psi}_k(i), & \text{if } \ell = k. \end{cases} \tag{11}$$

3) Projection. Each agent k projects the estimates $\{\boldsymbol{\psi}'_\ell(i)\}_{\ell \in \mathcal{N}_k}$ received from its neighborhood onto the linear manifold $\{w_{\mathcal{N}_k} : \mathcal{D}_k w_{\mathcal{N}_k} + b_k = \mathbf{0}_{j_k \times 1}\}$ to obtain

$$\boldsymbol{w}_k(i) = [\mathcal{P}_{\mathcal{N}_k}(i)]_k \operatorname{col}\{\boldsymbol{\psi}'_\ell(i)\}_{\ell \in \mathcal{N}_k} - [f_{\mathcal{N}_k}(i)]_k, \tag{12}$$

where the matrix $\mathcal{P}_{\mathcal{N}_k}(i)$ and vector $f_{\mathcal{N}_k}(i)$ are defined in (17) and (18), respectively, in the sequel.

Let $\delta = \operatorname{col}\{\delta_k\}_{k=1}^N$, where the non-negative numbers $\{\delta_k\}$ are the privacy thresholds in (7b). We call the proposed scheme (9), (11) and (12) *adapt-then-project* with privacy parameter δ or ATP(δ) in short. Specifically, we use ATP(0) algorithm to denote the case where there is no privacy constraint, *i.e.,* no additive noises are introduced in the exchange step in (11).

Remark 3: The differences between the proposed ATP(δ) algorithm (9) to (12) and the scheme in [7] are in the exchange and projection steps. Specifically, in order to protect each individual task $\boldsymbol{w}_k^o$ against privacy leakage, each agent k sends a noisy intermediate estimate $\boldsymbol{\psi}'_k(i)$, instead of the true estimate $\boldsymbol{\psi}_k(i)$ as in [7], to its neighboring agents. In addition, in the projection step (12), agent k projects its neighboring estimates onto the linear manifold corresponding to the intersection of all j_k linear equality constraints that it is involved in. In contrast, in [7], neighboring estimates of agent k are projected onto the linear manifold corresponding to each of the j_k constraints separately, which generates j_k intermediate estimates, and the new estimate is taken as the average of these intermediate estimates. Moreover, in order to mitigate the negative effect of privacy noises on the projection step, we introduce *weighted* projection operators in (14) in the sequel.

To allow a distributed implementation of the privacy mechanism, we make the following assumption.

Assumption 4. *(Privacy noise) The entries of $\boldsymbol{n}_k(i)$ at time instant i, for any $k = 1, \ldots, N$, are i.i.d. with zero mean and variance $\sigma^2_{n,k}(i)$, i.e.,*

$$R_{n,k}(i) \triangleq \mathbb{E}[\boldsymbol{n}_k(i)\boldsymbol{n}_k^\intercal(i)] = \sigma^2_{n,k}(i) I_{M_k}. \tag{13}$$

The random noises $\{\boldsymbol{n}_k(i)\}$ are white over time and space. The random process $\{\boldsymbol{n}_k(i)\}$ is independent of any other random processes.

Note that we use $\sigma^2_{v,k}$, defined in Assumption 1, to denote the variance of the measurement noise $\boldsymbol{v}_k(i)$ at agent k and all time instances i. We use $\sigma^2_{n,k}(i)$ for the time-varying variance of the privacy noise $\boldsymbol{n}_k(i)$ at agent k and time instant i. From Assumption 4, each agent k generates the noise $\boldsymbol{n}_k(i)$ independently of other agents in the network, and also independently over time instants $i \geq 0$. Based on this assumption, we now proceed to introduce the weighted projection operator $\mathcal{P}_{\mathcal{N}_k}(i)$ and vector $f_{\mathcal{N}_k}(i)$ involved in (12).

B. Weighted Projection Operator

For each agent k, let us collect the noisy intermediate estimates $\{\boldsymbol{\psi}'_\ell(i)\}$ defined by (11) from its neighboring agents $\{\ell \in \mathcal{N}_k\}$ into an $|\mathcal{N}_k| \times 1$ block column vector

$$\boldsymbol{\psi}'_{\mathcal{N}_k}(i) = \operatorname{col}\{\boldsymbol{\psi}'_\ell(i)\}_{\ell \in \mathcal{N}_k}.$$

Then, for each random realization $\boldsymbol{\psi}'_{\mathcal{N}_k}(i) = \psi'_{\mathcal{N}_k}(i)$, we are interested in seeking the optimal solution to the following optimization problem:

$$\min_{w_{\mathcal{N}_k}(i)} \ \|\psi'_{\mathcal{N}_k}(i) - w_{\mathcal{N}_k}(i)\|^2_{\Theta^{-1}_{\mathcal{N}_k}(i)} \tag{14a}$$

$$\text{s.t.} \ \ \mathcal{D}_k w_{\mathcal{N}_k}(i) + b_k = \mathbf{0}_{j_k \times 1}, \tag{14b}$$

where the weight matrix

$$\Theta^{-1}_{\mathcal{N}_k}(i) = \operatorname{diag}\{\theta_{\ell k}(i) I_{M_\ell}\}_{\ell \in \mathcal{N}_k}, \tag{15}$$

with

$$\theta_{\ell k}(i) > 0 \text{ for } \ell \in \mathcal{N}_k, \text{ and } \sum_{\ell \in \mathcal{N}_k} \theta_{\ell k}(i) = 1. \tag{16}$$

We note that $\Theta^{-1}_{\mathcal{N}_k}(i)$ is an $|\mathcal{N}_k| \times |\mathcal{N}_k|$ block-diagonal, symmetric and positive definite matrix. We assume that the weights $\{\theta_{\ell k}(i)\}$ defined by (15) converge as $i \to \infty$. Let

$$M_{\mathcal{N}_k} = \sum_{\ell \in \mathcal{N}_k} M_\ell,$$

and

$$\mathcal{P}_{\mathcal{N}_k}(i) = I_{M_{\mathcal{N}_k}} - \Theta_{\mathcal{N}_k}(i) \mathcal{D}_k^\intercal (\mathcal{D}_k \Theta_{\mathcal{N}_k}(i) \mathcal{D}_k^\intercal)^{-1} \mathcal{D}_k, \tag{17}$$

$$f_{\mathcal{N}_k}(i) = \Theta_{\mathcal{N}_k}(i) \mathcal{D}_k^\intercal (\mathcal{D}_k \Theta_{\mathcal{N}_k}(i) \mathcal{D}_k^\intercal)^{-1} b_k \tag{18}$$

where $\mathcal{P}_{\mathcal{N}_k}(i)$ is an $M_{\mathcal{N}_k} \times M_{\mathcal{N}_k}$ matrix and $f_{\mathcal{N}_k}(i)$ is of dimension $M_{\mathcal{N}_k} \times 1$. It follows that matrix $\mathcal{P}_{\mathcal{N}_k}(i)$ is a projection matrix. From [58], the minimizer of (14) is given by

$$w_{\mathcal{N}_k}(i) = \mathcal{P}_{\mathcal{N}_k}(i) \psi'_{\mathcal{N}_k}(i) - f_{\mathcal{N}_k}(i) \in \mathbb{R}^{M_{\mathcal{N}_k} \times 1}. \tag{19}$$

For each $\ell \in \mathcal{N}_k$, we are interested in the projection operator applied to the intermediate estimate from agent ℓ. This corresponds to an $M_k \times M_\ell$ block of $\mathcal{P}_{\mathcal{N}_k}(i)$, denoted as $[\mathcal{P}_{\mathcal{N}_k}(i)]_{k,\ell}$. We collect all these in an $N \times N$ block matrix $\mathcal{P}(i)$, whose $M_k \times M_\ell$ (k,ℓ)-th block equals

$$[\mathcal{P}(i)]_{k,\ell} = \begin{cases} [\mathcal{P}_{\mathcal{N}_k}(i)]_{k,\ell}, & \text{if } \ell \in \mathcal{N}_k, \\ \mathbf{0}_{M_k \times M_\ell}, & \text{otherwise.} \end{cases} \tag{20}$$

Similarly, we collect the block $[f_{\mathcal{N}_k}(i)]_k$ corresponding to agent k and define an $N \times 1$ block vector

$$f(i) = \operatorname{col}\{[f_{\mathcal{N}_k}(i)]_k\}_{k=1}^N. \tag{21}$$

C. Weight Matrix

The matrix $\Theta_{\mathcal{N}_k}^{-1}(i)$ is introduced in order to mitigate the negative effect of the privacy noises $\{\boldsymbol{n}_k(i)\}$ on the projection step (12). In the special case where

$$\Theta_{\mathcal{N}_k}^{-1}(i) = \operatorname{diag}\left\{\frac{1}{|\mathcal{N}_k|}I_{M_\ell}\right\}_{\ell\in\mathcal{N}_k},$$

the optimal solution of (14) is reduced to the Euclidean projection of $\psi'_{\mathcal{N}_k}(i)$ on the affine set $\{w_{\mathcal{N}_k} : \mathcal{D}_k w_{\mathcal{N}_k}+b_k = \mathbf{0}_{j_k\times 1}\}$ [59, p.398]. We proceed to rewrite (14a) as

$$\|\psi'_{\mathcal{N}_k}(i) - w_{\mathcal{N}_k}(i)\|^2_{\Theta^{-1}_{\mathcal{N}_k}(i)} = \sum_{\ell\in\mathcal{N}_k}\theta_{\ell k}(i)\|\psi'_\ell(i) - w_\ell(i)\|^2. \tag{22}$$

If the privacy noise power $\sigma^2_{n,\ell}(i)$ is large, a small weight $\theta_{\ell k}(i)$ should be assigned to the squared Euclidean distance $\|\psi'_\ell(i) - w_\ell(i)\|^2$ (see (104) in Section VI-A for a possible choice for the weights $\{\theta_{\ell k}(i)\}$). Thus in the extreme case where $\sigma^2_{n,\ell}(i)\to\infty$ for all $\ell\in\mathcal{N}_k\backslash\{k\}$, we have $\theta_{\ell k}(i)\to 0$ for all $\ell\in\mathcal{N}_k\backslash\{k\}$, and $\theta_{kk}(i)\to 1$. Then, from (22), the optimization problem in (14) is reduced to

$$\min_{w_{\mathcal{N}_k}(i)} \|\psi_k(i) - w_k(i)\|^2 \tag{23a}$$

$$\text{s.t.}\;\; \mathcal{D}_k w_{\mathcal{N}_k}(i) + b_k = \mathbf{0}_{j_k\times 1}. \tag{23b}$$

The optimal solution to (23) is then given by

$$w_k(i) = \psi_k(i),$$

and $\{w_\ell(i)\}$ for all $\ell\in\mathcal{N}_k\backslash\{k\}$ are chosen such that

$$\mathcal{D}_k w_{\mathcal{N}_k}(i) + b_k = \mathbf{0}_{j_k\times 1}.$$

In this case, the proposed ATP(δ) algorithm is reduced to the non-cooperative LMS algorithm, namely [60, p.165]:

$$\boldsymbol{w}_k(i) = \boldsymbol{w}_k(i-1) + \mu_k\boldsymbol{u}_k(i)\left(\boldsymbol{d}_k(i) - \boldsymbol{u}_k^{\mathsf{T}}(i)\boldsymbol{w}_k(i-1)\right). \tag{24}$$

D. Network Error Dynamics

To facilitate our analysis in later sections, we derive the network error dynamics in this subsection. Let

$$M = \sum_{k=1}^N M_k. \tag{25}$$

We collect the variables $\{\boldsymbol{w}_k^o\}, \{\boldsymbol{w}_k(i)\}$ and $\{\boldsymbol{\psi}_k(i)\}$ from all agents into the following $N\times 1$ block vectors:

$$\boldsymbol{w}^o = \operatorname{col}\{\boldsymbol{w}_k^o\}_{k=1}^N, \tag{26}$$

$$\boldsymbol{w}(i) = \operatorname{col}\{\boldsymbol{w}_k(i)\}_{k=1}^N, \tag{27}$$

$$\boldsymbol{\psi}(i) = \operatorname{col}\{\boldsymbol{\psi}_k(i)\}_{k=1}^N. \tag{28}$$

We also define the following network error vectors:

$$\widetilde{\boldsymbol{w}}(i) = \boldsymbol{w}^o - \boldsymbol{w}(i), \tag{29}$$

$$\widetilde{\boldsymbol{\psi}}(i) = \boldsymbol{w}^o - \boldsymbol{\psi}(i). \tag{30}$$

In addition, we define the following $N\times N$ block diagonal matrices:

$$\mathcal{M} = \operatorname{diag}\{\mu_k I_{M_k}\}_{k=1}^N, \tag{31}$$

$$\boldsymbol{\mathcal{R}}_u(i) = \operatorname{diag}\left\{\boldsymbol{u}_k(i)\boldsymbol{u}_k^{\mathsf{T}}(i)\right\}_{k=1}^N, \tag{32}$$

$$\mathcal{R}_u = \mathbb{E}[\boldsymbol{\mathcal{R}}_u(i)] = \operatorname{diag}\left\{R_{u,k}\right\}_{k=1}^N. \tag{33}$$

We collect the privacy noises $\{\boldsymbol{n}_k(i)\}$ from all agents into an $N\times 1$ block column vector $\boldsymbol{n}(i)$ and define its covariance matrix

$$\boldsymbol{n}(i) = \operatorname{col}\{\boldsymbol{n}_k(i)\}_{k=1}^N, \tag{34}$$

$$R_n(i) = \mathbb{E}[\boldsymbol{n}(i)\boldsymbol{n}^{\mathsf{T}}(i)] = \operatorname{diag}\left\{R_{n,k}(i)\right\}_{k=1}^N. \tag{35}$$

We also define the following $N\times 1$ block column vector and its covariance matrix

$$\boldsymbol{g}(i) = \operatorname{col}\{\boldsymbol{u}_k(i)\boldsymbol{v}_k(i)\}_{k=1}^N, \tag{36}$$

$$\mathcal{G} = \mathbb{E}[\boldsymbol{g}(i)\boldsymbol{g}^{\mathsf{T}}(i)] = \operatorname{diag}\left\{R_{u,k}\sigma^2_{v,k}\right\}_{k=1}^N. \tag{37}$$

Let $[\mathcal{P}(i)]_{k^-}$ be the k-th block row of $\mathcal{P}(i)$ defined by (20) by setting $[\mathcal{P}(i)]_{k,k} = \mathbf{0}_{M_k\times M_k}$,

$$\boldsymbol{q}_k(i) = [\mathcal{P}(i)]_{k^-}\boldsymbol{n}(i) \in \mathbb{R}^{M_k\times 1}, \tag{38}$$

$$\boldsymbol{q}(i) = \operatorname{col}\{\boldsymbol{q}_k(i)\}_{k=1}^N \in \mathbb{R}^{M\times 1}. \tag{39}$$

We also define an $N\times N$ block matrix

$$\Gamma(i) = \mathbb{E}[\boldsymbol{q}(i)\boldsymbol{q}^{\mathsf{T}}(i)] \in \mathbb{R}^{M\times M},$$

whose $M_k\times M_\ell$ (k,ℓ)-th block entry equals

$$\begin{aligned}[\Gamma(i)]_{k,\ell} &= \mathbb{E}[\boldsymbol{q}_k(i)\boldsymbol{q}_\ell^{\mathsf{T}}(i)]\\ &\overset{(38)}{=} \mathbb{E}\left[[\mathcal{P}(i)]_{k^-}\boldsymbol{n}(i)\boldsymbol{n}^{\mathsf{T}}(i)[\mathcal{P}(i)]_{\ell^-}^{\mathsf{T}}\right]\\ &\overset{(35)}{=} [\mathcal{P}(i)]_{k^-}R_n(i)[\mathcal{P}(i)]_{\ell^-}^{\mathsf{T}}.\end{aligned} \tag{40}$$

We have the following recursion for the network error vector $\widetilde{\boldsymbol{w}}(i)$.

Lemma 1. *Consider the distributed strategy (9) to (12). The evolution of the error dynamics across the network relative to the reference vector $\boldsymbol{w}^o$ defined by (26) is described by the following recursion:*

$$\widetilde{\boldsymbol{w}}(i) = \mathcal{P}(i)(I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i))\widetilde{\boldsymbol{w}}(i-1) - \mathcal{P}(i)\mathcal{M}\boldsymbol{g}(i) - \boldsymbol{q}(i), \tag{41}$$

$$\begin{aligned}\widetilde{\boldsymbol{\psi}}(i+1) = {}&(I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1))\mathcal{P}(i)\widetilde{\boldsymbol{\psi}}(i)\\ &- (I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1))\boldsymbol{q}(i) - \mathcal{M}\boldsymbol{g}(i+1)\end{aligned} \tag{42}$$

for any time instant $i\geq 0$.

Proof. From (9), we have

$$\widetilde{\boldsymbol{\psi}}(i) = (I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i))\widetilde{\boldsymbol{w}}(i-1) - \mathcal{M}\boldsymbol{g}(i). \tag{43}$$

From (11) and (12), we obtain

$$\boldsymbol{w}(i) = \mathcal{P}(i)\boldsymbol{\psi}(i) + \boldsymbol{q}(i) - f(i). \tag{44}$$

Since the parameter vectors $\{\boldsymbol{w}_\ell^o\}_{\ell\in\mathcal{N}_k}$ satisfy the local constraints (5) at agent k, we have

$$\boldsymbol{w}^o = \mathcal{P}(i)\boldsymbol{w}^o - f(i). \tag{45}$$

Subtracting (44) from both sides of (45), we have

$$\widetilde{\boldsymbol{w}}(i) = \mathcal{P}(i)\widetilde{\boldsymbol{\psi}}(i) - \boldsymbol{q}(i). \tag{46}$$

Substituting (43) into the right-hand side (R.H.S.) of (46), we arrive at the desired recursion (41). Substituting (46) into the R.H.S. of (43), we obtain the desired recursion (42), and the proof is complete. □

Let Σ be a symmetric positive semi-definite matrix, and

$$\Sigma'(i) = \mathbb{E}[(I_M - \boldsymbol{\mathcal{R}}_u(i)\mathcal{M})\mathcal{P}^{\mathsf{T}}(i)\Sigma\mathcal{P}(i)(I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i))]. \tag{47}$$

From (41), we have

$$\begin{aligned}
\mathbb{E}\|\widetilde{\boldsymbol{w}}(i)\|^2_{\Sigma} &\overset{\text{(a)}}{=} \mathbb{E}\|\widetilde{\boldsymbol{w}}(i-1)\|^2_{\Sigma'(i)} + \mathbb{E}[\boldsymbol{q}^{\mathsf{T}}(i)\Sigma\boldsymbol{q}(i)] \\
&\quad + \mathbb{E}[\boldsymbol{g}^{\mathsf{T}}(i)\mathcal{M}\mathcal{P}^{\mathsf{T}}(i)\Sigma\mathcal{P}(i)\mathcal{M}\boldsymbol{g}(i)] \\
&\overset{\text{(b)}}{=} \mathbb{E}\|\widetilde{\boldsymbol{w}}(i-1)\|^2_{\Sigma'(i)} + \mathbb{E}[\mathrm{Tr}(\boldsymbol{q}(i)\boldsymbol{q}^{\mathsf{T}}(i)\Sigma)] \\
&\quad + \mathbb{E}[\mathrm{Tr}(\boldsymbol{g}(i)\boldsymbol{g}^{\mathsf{T}}(i)\mathcal{M}\mathcal{P}^{\mathsf{T}}(i)\Sigma\mathcal{P}(i)\mathcal{M})] \\
&\overset{\text{(c)}}{=} \mathbb{E}\|\widetilde{\boldsymbol{w}}(i-1)\|^2_{\Sigma'(i)} + \mathrm{Tr}(\Gamma(i)\Sigma) \\
&\quad + \mathrm{Tr}(\mathcal{G}\mathcal{M}\mathcal{P}^{\mathsf{T}}(i)\Sigma\mathcal{P}(i)\mathcal{M}),
\end{aligned} \tag{48}$$

where in (a) we used Assumptions 1 to 4, and

$$\mathbb{E}\boldsymbol{g}(i) = \boldsymbol{0}_{M\times 1}, \tag{49}$$

$$\mathbb{E}\boldsymbol{q}(i) = \boldsymbol{0}_{M\times 1}, \tag{50}$$

in (b) we used the property $\mathrm{Tr}(AB) = \mathrm{Tr}(BA)$ for any matrices $\{A, B\}$ of compatible sizes, and in (c) we interchanged expectation and trace. We will rely on the error dynamics as shown by Lemma 1 and (48) to design the privacy noise power $\sigma^2_{n,k}(i)$ at each agent k and time instant i in Section IV, and to evaluate the mean behavior and mean-square performance of the proposed ATP(δ) algorithm (9) to (12) in Section V in the sequel.

IV. Privacy Noise Design

In this section, we present an approximate solution to the utility-privacy optimization trade-off problem (7) by first deriving a sufficient condition that leads to the privacy constraints. Finally, we motivate a distributed and adaptive scheme for each agent to compute its local privacy noise power at each time instant.

A. Privacy Noise Power

We start by showing that the quantity $\mathrm{MSD}_{\mathrm{net}}(i)$ is a monotonically increasing function with respect to (w.r.t.) the privacy noise powers $\left\{\sigma^2_{n,k}(i)\right\}$.

Lemma 2. *Consider the distributed strategy (9) to (12). Suppose that Assumptions 1 to 4 hold. Then, $\mathrm{MSD}_{net}(i)$ is an increasing function w.r.t. the privacy noise powers $\left\{\sigma^2_{n,k}(i)\right\}$.*

Proof. From (7a), we have

$$\mathrm{MSD}_{\mathrm{net}}(i) = \mathbb{E}\|\widetilde{\boldsymbol{w}}(i)\|^2_{I_M/N}. \tag{51}$$

By setting

$$\Sigma = \frac{1}{N} I_M \tag{52}$$

on both sides of (48), we observe that the error $\widetilde{\boldsymbol{w}}(i-1)$ on the R.H.S. of (48) only relies on the random variables $\{\boldsymbol{d}_k(j), \boldsymbol{u}_k(j), \boldsymbol{n}_k(j) \,:\, k = 1, \ldots, N,\ 0 \le j \le i-1\}$, thus its value is not affected by the variances $\{\sigma^2_{n,k}(i)\}$ at time instant i. The value of $\mathrm{MSD}_{\mathrm{net}}(i)$ depends on the variances $\{\sigma^2_{n,k}(i)\}$ via the matrix $\Gamma(i)$ in (40). We have

$$\begin{aligned}
\mathrm{Tr}(\Gamma(i)\Sigma) &\overset{(52)}{=} \frac{1}{N}\mathrm{Tr}(\Gamma(i)) \\
&= \frac{1}{N}\sum_{k=1}^{N}\mathrm{Tr}([\Gamma(i)]_{k,k})
\end{aligned} \tag{53}$$

and

$$\begin{aligned}
\mathrm{Tr}([\Gamma(i)]_{k,k}) &\overset{(40)}{=} \mathrm{Tr}\Big(\sum_{\ell\in\mathcal{N}_k\setminus\{k\}} [\mathcal{P}(i)]_{k,\ell} R_{n,\ell} \,[\mathcal{P}(i)]^{\mathsf{T}}_{k,\ell}\Big) \\
&= \sum_{\ell\in\mathcal{N}_k\setminus\{k\}} \mathrm{Tr}([\mathcal{P}(i)]_{k,\ell} R_{n,\ell} \,[\mathcal{P}(i)]^{\mathsf{T}}_{k,\ell}) \\
&= \sum_{\ell\in\mathcal{N}_k\setminus\{k\}} \mathrm{Tr}(R_{n,\ell} \,[\mathcal{P}(i)]^{\mathsf{T}}_{k,\ell}[\mathcal{P}(i)]_{k,\ell}) \\
&\overset{(13)}{=} \sum_{\ell\in\mathcal{N}_k\setminus\{k\}} \sigma^2_{n,\ell}(i)\,\mathrm{Tr}([\mathcal{P}(i)]^{\mathsf{T}}_{k,\ell}[\mathcal{P}(i)]_{k,\ell}),
\end{aligned} \tag{54}$$

which completes the proof. □

In view of Lemma 2, solving the optimization problem (7) is reduced to finding the smallest privacy noise powers that satisfy the privacy constraints (7b). For each agent $k = 1, \ldots, N$, let

$$U_{kk}(i) = \mathbb{E}[(\boldsymbol{w}^o_k - \mathbb{E}\boldsymbol{w}^o_k)(\boldsymbol{\psi}'_k(i) - \mathbb{E}\boldsymbol{\psi}'_k(i))^{\mathsf{T}}], \tag{55}$$

$$X_{kk}(i) = \mathbb{E}[(\boldsymbol{\psi}_k(i) - \mathbb{E}\boldsymbol{\psi}_k(i))\,(\boldsymbol{\psi}_k(i) - \mathbb{E}\boldsymbol{\psi}_k(i))^{\mathsf{T}}], \tag{56}$$

$$X'_{kk}(i) = \mathbb{E}\big[(\boldsymbol{\psi}'_k(i) - \mathbb{E}\boldsymbol{\psi}'_k(i))\,(\boldsymbol{\psi}'_k(i) - \mathbb{E}\boldsymbol{\psi}'_k(i))^{\mathsf{T}}\big], \tag{57}$$

all of which are of dimension $M_k \times M_k$, and the $M_k \times 1$ vector

$$\widehat{\boldsymbol{w}}_{k|\boldsymbol{\psi}'_k}(i) = U_{kk}(i)\,(X'_{kk}(i))^{-1}\,(\boldsymbol{\psi}'_k(i) - \mathbb{E}\boldsymbol{\psi}'_k(i)) + \mathbb{E}\boldsymbol{w}^o_k \tag{58}$$

be the linear least-mean-square estimator [60, p.66] of $\boldsymbol{w}^o_k$ at time instant i, given $\boldsymbol{\psi}'_k(i)$ (note that $\widehat{\boldsymbol{w}}_{k|\boldsymbol{\psi}'_k}(i)$ and $\boldsymbol{\psi}'_k(i)$ are linearly related in our ATP(δ) strategy).

From Assumption 4, we have

$$X'_{kk}(i) = X_{kk}(i) + R_{n,k}(i) = X_{kk}(i) + \sigma^2_{n,k}(i) I_{M_k} \tag{59}$$

and using [60, p.66], we obtain

$$\begin{aligned}
&\mathbb{E}\|\boldsymbol{w}^o_k - \widehat{\boldsymbol{w}}_{k|\boldsymbol{\psi}'_k}(i)\|^2 \\
&\quad = \mathrm{Tr}(W_{kk} - U_{kk}(i)\,(X'_{kk}(i))^{-1}\,U^{\mathsf{T}}_{kk}(i)) \\
&\quad = \mathrm{Tr}(W_{kk}) - \mathrm{Tr}(U_{kk}(i)\,(X'_{kk}(i))^{-1}\,U^{\mathsf{T}}_{kk}(i)),
\end{aligned} \tag{60}$$

where the matrix W_{kk} is defined in (2). Substituting (60) into the left-hand side (L.H.S.) of the privacy constraint (7b), we have

$$\begin{aligned}
&\mathrm{Tr}(U_{kk}(i)\,\big(X_{kk}(i) + \sigma^2_{n,k}(i) I_{M_k}\big)^{-1}\,U^{\mathsf{T}}_{kk}(i)) \\
&\qquad\qquad \le \mathrm{Tr}(W_{kk}) - \delta_k.
\end{aligned} \tag{61}$$

A numerical solution for the optimal noise powers $\{\sigma_{n,k}^2(i)\}$ can be obtained by seeking the smallest values that satisfy the inequality (61). However, in view of the fact that agents in the network work in a cooperative manner, the evaluation of $\{U_{kk}(i), X_{kk}(i)\}$ involves global statistics that are not available locally at each agent k. In this paper, we are interested in a *distributed and adaptive* scheme to compute the noise power $\sigma_{n,k}^2(i)$ for each agent k and time instant $i \geq 0$. To this end, we first derive a closed-form expression for a noise power $\widehat{\sigma}_{n,k}^2(i)$ that satisfies the privacy constraint (7b) for each agent k and time instant $i \geq 0$.

Theorem 1. *Consider the distributed strategy (9) to (12). Suppose that Assumptions 1 to 4 hold. If*

$$\sigma_{n,k}^2(i) \geq \frac{\mathrm{Tr}(U_{kk}^{\mathsf{T}}(i)U_{kk}(i))}{\mathrm{Tr}(W_{kk}) - \delta_k} \tag{62}$$

for each agent k and time instant $i \geq 0$, then the privacy constraint (7b) is satisfied.

Proof. See Appendix A. □

In view of Theorem 1 and Lemma 2, to reduce $\mathrm{MSD}_{\mathrm{net}}(i)$ as much as possible, we set the privacy noise power of agent k at time instant $i \geq 0$ to be

$$\widehat{\sigma}_{n,k}^2(i) = \frac{\mathrm{Tr}(U_{kk}^{\mathsf{T}}(i)U_{kk}(i))}{\mathrm{Tr}(W_{kk}) - \delta_k}. \tag{63}$$

We note that this is a feasible solution to (7) due to Theorem 1 and moreover $\widehat{\sigma}_{n,k}^2(i) > 0$ because $\mathrm{Tr}(U_{kk}^{\mathsf{T}}(i)U_{kk}(i)) > 0$ and (8). We next show that the sequence $\left\{\widehat{\sigma}_{n,k}^2(i)\right\}_{i\geq 0}$ defined by (63) is convergent for any $k = 1, \ldots, N$. To facilitate our analysis, we make the following assumption.

Assumption 5. *The matrix $[\mathcal{D}_{k_o}]_{\cdot,k_o}$ is column-rank deficient for at least one agent k_o.*

This assumption holds true for sufficiently large M_{k_o} such that $\mathrm{rank}(\mathcal{D}_{k_o}) < M_{k_o}$ holds.

Lemma 3. *Consider the matrix $\mathcal{P}(i)$ defined by (20). Suppose that Assumption 5 holds. Suppose also that the matrices $\{\Theta_{\mathcal{N}_k}(i)\}$ are positive-definite for all agents k and time instants $i \geq 0$. For any time instant $i \geq 0$, we have*

$$\|\mathcal{P}(i)\| \geq 1. \tag{64}$$

Proof. Let ψ be an $N \times 1$ block vector, with each block entry of size $M_k \times 1$. We choose $\psi = \left[0^{\mathsf{T}}, \ldots, 0^{\mathsf{T}}, \psi_{k_o}^{\mathsf{T}}, 0^{\mathsf{T}}, \ldots, 0^{\mathsf{T}}\right]^{\mathsf{T}}$ with a non-zero vector at the k_o-th block for some $k_o = 1, \ldots, N$. We have

$$\begin{aligned}\|\mathcal{P}(i)\psi\|^2 &= \sum_{k=1}^{N} \|[\mathcal{P}(i)]_k \psi\|^2 \\ &\geq \|[\mathcal{P}(i)]_{k_o} \psi\|^2 \\ &= \|[\mathcal{P}(i)]_{k_o,k_o} \psi_{k_o}\|^2. \end{aligned} \tag{65}$$

From (17) and (20), we obtain

$$\begin{aligned}&[\mathcal{P}(i)]_{k_o,k_o} \\ &= \left[\mathcal{P}_{\mathcal{N}_{k_o}}(i)\right]_{k_o,k_o} \\ &= \left[I_{M_{\mathcal{N}_{k_o}}} - \Theta_{\mathcal{N}_{k_o}}(i)\mathcal{D}_{k_o}^{\mathsf{T}}(\mathcal{D}_{k_o}\Theta_{\mathcal{N}_{k_o}}(i)\mathcal{D}_{k_o}^{\mathsf{T}})^{-1}\mathcal{D}_{k_o}\right]_{k_o,k_o} \\ &= I_{M_{k_o}} - \left[\Theta_{\mathcal{N}_{k_o}}(i)\mathcal{D}_{k_o}^{\mathsf{T}}\right]_{k_o}(\mathcal{D}_{k_o}\Theta_{\mathcal{N}_{k_o}}(i)\mathcal{D}_{k_o}^{\mathsf{T}})^{-1}[\mathcal{D}_{k_o}]_{\cdot,k_o} \\ &\overset{(15)}{=} I_{M_{k_o}} - \frac{1}{\theta_{k_o,k_o}(i)}\left[\mathcal{D}_{k_o}^{\mathsf{T}}\right]_{k_o}(\mathcal{D}_{k_o}\Theta_{\mathcal{N}_{k_o}}(i)\mathcal{D}_{k_o}^{\mathsf{T}})^{-1}[\mathcal{D}_{k_o}]_{\cdot,k_o}.\end{aligned}$$

Note that the matrix $\left[\mathcal{D}_{k_o}^{\mathsf{T}}\right]_{k_o}(\mathcal{D}_{k_o}\Theta_{\mathcal{N}_{k_o}}(i)\mathcal{D}_{k_o}^{\mathsf{T}})^{-1}[\mathcal{D}_{k_o}]_{\cdot,k_o}$ is positive semi-definite, and it has a zero eigenvalue given Assumption 5. Therefore $[\mathcal{P}(i)]_{k_o,k_o}$ has an eigenvalue 1. In view of the fact that the $[\mathcal{P}(i)]_{k_o,k_o}$ is a symmetric matrix, we conclude that

$$\|[\mathcal{P}(i)]_{k_o,k_o}\| \geq 1.$$

Thus there exists ψ_{k_o} such that

$$\|[\mathcal{P}(i)]_{k_o,k_o}\psi_{k_o}\| \geq \|\psi_{k_o}\| = \|\psi\|. \tag{66}$$

It then follows from (65) and (66) that

$$\|\psi\| \leq \|\mathcal{P}(i)\psi\| \leq \|\mathcal{P}(i)\|\|\psi\|,$$

and the proof is complete. □

Note that Lemma 3 helps to simplify the upper bounds in Proposition 1 and Theorem 2 in the sequel.

Proposition 1. *Consider the distributed strategy (9) to (12). Suppose that Assumptions 1 to 5 hold. Suppose also that*

$$\frac{1 - 1/\|\mathcal{P}(i)\|}{1 + 1/\|\mathcal{P}(i)\|} < \frac{\lambda_{\min}(R_{u,k})}{\lambda_{\max}(R_{u,k})} \leq 1$$

for all agents k. If the step-size μ_k satisfies

$$\frac{1 - 1/\|\mathcal{P}(i)\|}{\lambda_{\min}(R_{u,k})} < \mu_k < \frac{1 + 1/\|\mathcal{P}(i)\|}{\lambda_{\max}(R_{u,k})} \tag{67}$$

for each agent k, then

$$\lim_{i\to\infty} \widehat{\sigma}_{n,k}^2(i) = \frac{\mathrm{Tr}(W_{kk}^2)}{\mathrm{Tr}(W_{kk}) - \delta_k}, \tag{68}$$

where $\widehat{\sigma}_{n,k}^2(i)$ is defined by (63)*.*

Proof. The proof involves mainly algebraic manipulations and is provided in Section S1 of the supplementary material. □

We next motivate a scheme based on (68) for agents in the network to compute the privacy noise powers in a distributed and adaptive manner.

B. Distributed and Adaptive ATP(δ)

The covariance matrix W_{kk} may not be known beforehand, and may change over time. In this subsection, we motivate a distributed and adaptive scheme that enables each agent k to estimate $\widehat{\sigma}_{n,k}^2(\infty)$ defined by (68) locally by using data available to it at time instant i. We assume that the statistics $\mathbb{E}\boldsymbol{w}_k^o$ is available at each agent k *a priori*; otherwise it can be estimated via

$$\mathbb{E}\boldsymbol{w}_k^o = R_{u,k}^{-1}\,\mathbb{E}[\boldsymbol{d}_k(i)\boldsymbol{u}_k(i)]$$

which follows from (1) and Assumptions 2 and 3, and where the quantities $\{R_{u,k}, \mathbb{E}[\boldsymbol{d}_k(i)\boldsymbol{u}_k(i)]\}$ can be inferred from the random data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_k(i)\}$. Let $\boldsymbol{\beta}_k(i)$ be an estimate of $\mathrm{Tr}(W_{kk}^2)$, and $\boldsymbol{\gamma}_k(i)$ an estimate of $\mathrm{Tr}(W_{kk})$ respectively at time instant i. We start with $\boldsymbol{\beta}_k(-1) = 0$ and $\boldsymbol{\gamma}_k(-1) = \delta_k$ for every agent k. For any time instant $i \geq 0$, we compute the quantities $\{\boldsymbol{\beta}_k(i), \boldsymbol{\gamma}_k(i)\}$ at each agent k using

$$\begin{aligned}
\boldsymbol{R}_{\psi,k}(i) &= (\boldsymbol{\psi}_k(i) - \mathbb{E}\boldsymbol{w}_k^o)(\boldsymbol{\psi}_k(i) - \mathbb{E}\boldsymbol{w}_k^o)^{\mathsf{T}} \in \mathbb{R}^{M_k \times M_k}, \\
\boldsymbol{\beta}_k(i) &= \alpha\boldsymbol{\beta}_k(i-1) + (1-\alpha)\,\mathrm{Tr}(\boldsymbol{R}_{\psi,k}^2(i)), \\
\boldsymbol{\gamma}_k(i) &= \alpha\boldsymbol{\gamma}_k(i-1) + (1-\alpha)\,\mathrm{Tr}(\boldsymbol{R}_{\psi,k}(i)),
\end{aligned}$$

where $\boldsymbol{\psi}_k(i)$ is the intermediate estimate generated by (9), and the parameter $0 < \alpha < 1$ is a forgetting factor. Then, agent k estimates $\widehat{\sigma}_{n,k}^2(\infty)$ using

$$\widehat{\widehat{\boldsymbol{\sigma}}}_{n,k}^2(i) = \begin{cases} \alpha\widehat{\widehat{\boldsymbol{\sigma}}}_{n,k}^2(i-1) + (1-\alpha)\frac{\boldsymbol{\beta}_k(i)}{\boldsymbol{\gamma}_k(i)-\delta_k}, \\ \qquad\qquad \text{if } \boldsymbol{\beta}_k(i)/(\boldsymbol{\gamma}_k(i) - \delta_k) > 0, \\ \widehat{\widehat{\boldsymbol{\sigma}}}_{n,k}^2(i-1), \qquad\qquad \text{otherwise}, \end{cases} \tag{69}$$

where $\widehat{\widehat{\boldsymbol{\sigma}}}_{n,k}^2(-1) = 0$. A privacy noise $\boldsymbol{n}_k(i)$ is generated at agent k by following a distribution with zero mean and variance $\widehat{\widehat{\boldsymbol{\sigma}}}_{n,k}^2(i)$. This noise is added to the intermediate estimate $\boldsymbol{\psi}_k(i)$ to form the noisy estimate $\boldsymbol{\psi}_k'(i)$. The quantity $\boldsymbol{\psi}_k'(i)$ is then transmitted to the neighboring agents $\{\ell \in \mathcal{N}_k \backslash \{k\}\}$. Fig. 1 summarizes our proposed distributed and adaptive ATP(δ) algorithm.

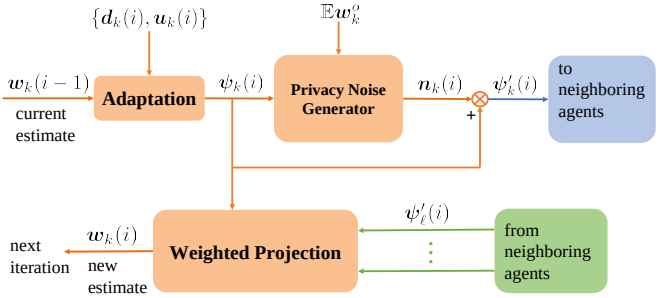


Fig. 1. Diagram of the proposed distributed and adaptive ATP(δ) algorithm.

V. Performance Analysis

In this section, we study the mean and mean-square behaviors of the proposed ATP(δ) algorithm, where the privacy noise power at each agent k and time instant $i \geq 0$, $\sigma_{n,k}^2(i)$, is set to $\widehat{\sigma}_{n,k}^2(i)$ defined in (63). For simplicity, we denote this setting by $\left\{\sigma_{n,k}^2(i) = \widehat{\sigma}_{n,k}^2(i)\right\}$. In addition, we evaluate its privacy-preserving performance.

A. Mean Behavior Analysis

We have the following theorem for the network mean performance.

Theorem 2. *Consider the distributed strategy (9) to (12) with privacy noise powers $\left\{\sigma_{n,k}^2(i) = \widehat{\sigma}_{n,k}^2(i)\right\}$. Suppose that Assumptions 1 to 5 hold, and*

$$\frac{1 - 1/\|\mathcal{P}(i)\|}{1 + 1/\|\mathcal{P}(i)\|} < \frac{\lambda_{\min}(R_{u,k})}{\lambda_{\max}(R_{u,k})} \leq 1$$

for all agents $k = 1, \ldots, N$. If the step-size μ_k is chosen to satisfy

$$\frac{1 - 1/\|\mathcal{P}(i)\|}{\lambda_{\min}(R_{u,k})} < \mu_k < \frac{1 + 1/\|\mathcal{P}(i)\|}{\lambda_{\max}(R_{u,k})} \tag{70}$$

for each agent k, then

$$\lim_{i\to\infty} \mathbb{E}\widetilde{\boldsymbol{w}}(i) = 0. \tag{71}$$

Proof. Let

$$\mathcal{A}(i) \triangleq \mathcal{P}(i)(I_M - \mathcal{M}\mathcal{R}_u) \in \mathbb{R}^{M\times M}. \tag{72}$$

Taking expectations on both sides of (41) gives

$$\begin{aligned}
\mathbb{E}\widetilde{\boldsymbol{w}}(i) &= \mathcal{P}(i)\mathbb{E}[(I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i))\widetilde{\boldsymbol{w}}(i-1)] - \mathcal{P}(i)\mathcal{M}\mathbb{E}\boldsymbol{g}(i) \\
&\quad - \mathbb{E}\boldsymbol{q}(i) \\
&\overset{\text{(a)}}{=} \mathcal{A}(i)\mathbb{E}\widetilde{\boldsymbol{w}}(i-1) - \mathcal{P}(i)\mathcal{M}\mathbb{E}\boldsymbol{g}(i) - \mathbb{E}\boldsymbol{q}(i) \\
&\overset{\text{(b)}}{=} \mathcal{A}(i)\mathbb{E}\widetilde{\boldsymbol{w}}(i-1)
\end{aligned} \tag{73}$$

where in step (a) we used Assumption 2, and step (b) follows from Assumptions 1, 3 and 4. Using the same proof as in Lemma S1.1 of the supplementary material, $\rho(\mathcal{A}(i)) < 1$ if (70) holds. The proof is now complete. □

By recalling the definition of $\widetilde{\boldsymbol{w}}(i)$ in (29), we observe from Theorem 2 that at each agent k in the network, for each realization w_k^o of the unknown random parameter vector $\boldsymbol{w}_k^o$, the proposed ATP(δ) algorithm (9) to (12) with privacy noise powers $\left\{\sigma_{n,k}^2(i) = \widehat{\sigma}_{n,k}^2(i)\right\}$ provides an *asymptotically unbiased* or consistent estimator $\boldsymbol{w}_k(i)$ as $i \to \infty$.

B. Mean-square Performance Analysis

In this subsection, we evaluate the steady-state network MSD, namely, the value of $\mathrm{MSD}_{\mathrm{net}}(i)$ as the time instant $i \to \infty$. Let

$$\begin{aligned}
\mathcal{F}(i) = \mathbb{E}\,[\,&((I_M - \boldsymbol{\mathcal{R}}_u(i)\mathcal{M})\mathcal{P}^{\mathsf{T}}(i)) \\
&\otimes ((I_M - \boldsymbol{\mathcal{R}}_u(i)\mathcal{M})\mathcal{P}^{\mathsf{T}}(i))\,] \in \mathbb{R}^{M^2\times M^2},
\end{aligned} \tag{74}$$

where $\otimes$ denotes Kronecker product.

Assumption 6. *For any $k = 1, \ldots, N$, the noise powers $\left\{\sigma_{n,k}^2(i)\right\}_{i\geq 0}$ converge as $i \to \infty$.*

We note that Assumption 6 is satisfied if we set $\sigma_{n,k}^2(i) = \widehat{\sigma}_{n,k}^2(i)$ for all agents $k = 1, \ldots, N$ and $i \geq 0$. Assumption 6 together with the assumption that $\theta_{\ell k}(i)$ defined by (16) converges as $i \to \infty$ for any agents $\ell, k = 1, \cdots, N$ lead to the following conclusions:

(a) $\Gamma(i)$ converges as $i \to \infty$.
(b) $\mathcal{F} = \lim_{i\to\infty} \mathcal{F}(i)$ exists.

In addition, we assume that the quantity $(1-1/\|\mathcal{P}(i)\|)/\lambda_{\min}(R_{u,k})$ on the L.H.S. of (70) is sufficiently small, so that for sufficiently small step-sizes $\{\mu_k\}$, we have

$$\mathcal{F}=\mathcal{A}^{\mathsf{T}}(\infty)\otimes\mathcal{A}^{\mathsf{T}}(\infty)+O(\mu_{\max}^2)\in\mathbb{R}^{M^2\times M^2}$$

where $\mathcal{A}(\infty)\triangleq\lim_{i\to\infty}\mathcal{A}(i)$ with $\mathcal{A}(i)$ defined by (72), and $\mu_{\max}\triangleq\max\{\mu_1,\ldots,\mu_N\}$. Then if the step-sizes $\{\mu_k\}$ satisfy (70), we have $\rho(\mathcal{A}(\infty))<1$, and $\rho(\mathcal{F})\approx(\rho(\mathcal{A}(\infty)))^2<1$, i.e., the matrix $\mathcal{F}$ is stable. Let

$$\sigma_{\rm ss}=\frac{1}{N}\left(I_{M^2}-\mathcal{F}\right)^{-1}\mathrm{vec}\left(I_M\right). \tag{75}$$

Theorem 3. *Consider the distributed strategy (9) to (12). Suppose that Assumptions 1 to 6 hold. Then,*

$$\begin{aligned}\textit{MSD}_{\textit{net}}&\triangleq\lim_{i\to\infty}\textit{MSD}_{\textit{net}}(i)\\&=\lim_{i\to\infty}\left[\mathrm{vec}\left(\mathcal{P}(i)\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{P}^{\mathsf{T}}(i)\right)\right]^{\mathsf{T}}\sigma_{ss}\\&\quad+\lim_{i\to\infty}\left[\mathrm{vec}\left(\Gamma(i)\right)\right]^{\mathsf{T}}\sigma_{ss}.\end{aligned} \tag{76}$$

Proof. Note that [51, p.762]

$$\mathrm{Tr}(AB)=\left[\mathrm{vec}(B^{\mathsf{T}})\right]^{\mathsf{T}}\mathrm{vec}(A) \tag{77}$$

$$\mathrm{vec}(ACB)=\left(B^{\mathsf{T}}\otimes A\right)\mathrm{vec}(C) \tag{78}$$

for any matrices $\{A,B,C\}$ of compatible sizes. Let $\sigma=\mathrm{vec}(\Sigma)$. Then, it follows from (78) that

$$\sigma'(i)\triangleq\mathrm{vec}\left(\Sigma'(i)\right)=\mathcal{F}(i)\sigma.$$

Now, we rewrite the recursion (48) as

$$\begin{aligned}\mathbb{E}\|\widetilde{\boldsymbol{w}}(i)\|_{\sigma}^2&=\mathbb{E}\|\widetilde{\boldsymbol{w}}(i-1)\|_{\mathcal{F}(i)\sigma}^2+\left[\mathrm{vec}\left(\Gamma(i)\right)\right]^{\mathsf{T}}\sigma\\&\quad+\left[\mathrm{vec}\left(\mathcal{P}(i)\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{P}^{\mathsf{T}}(i)\right)\right]^{\mathsf{T}}\sigma,\end{aligned} \tag{79}$$

where we used the property (77). From Assumption 6, we obtain

$$\begin{aligned}\lim_{i\to\infty}\mathbb{E}\|\widetilde{\boldsymbol{w}}(i)\|_{\left(I_{M^2}-\mathcal{F}\right)\sigma}^2&=\lim_{i\to\infty}\left[\mathrm{vec}\left(\mathcal{P}(i)\mathcal{M}\mathcal{G}\mathcal{M}\mathcal{P}^{\mathsf{T}}(i)\right)\right]^{\mathsf{T}}\sigma\\&\quad+\lim_{i\to\infty}\left[\mathrm{vec}\left(\Gamma(i)\right)\right]^{\mathsf{T}}\sigma.\end{aligned}$$

By setting $\sigma=\sigma_{\rm ss}$, which is defined by (75) on the both sides, we arrive at the desired result (76) and the proof is complete. □

C. Privacy-preserving Performance

In this subsection, we introduce the performance metrics for inference privacy preservation, and evaluate the privacy-preserving performance of the proposed ATP(δ) algorithm.

Let $\boldsymbol{\Phi}_\ell(i)$ be the set of information that an agent ℓ has at time instant $i\geq 0$ to infer its neighboring agent k's task $\boldsymbol{w}_k^o$. To quantify the privacy-preserving performance of any scheme, we evaluate, in terms of mean-square error, how well agent ℓ can estimate $\boldsymbol{w}_k^o$. Then we average over the whole network. The network inference privacy error is defined using

$$\xi_{\rm net}(i)=\frac{1}{N}\sum_{k=1}^{N}\frac{1}{|\mathcal{N}_k|-1}\sum_{\ell\in\mathcal{N}_k\setminus\{k\}}\mathbb{E}\|\boldsymbol{w}_k^o-\widehat{\boldsymbol{w}}_{k|\boldsymbol{\Phi}_\ell}(i)\|^2, \tag{80}$$

where $\widehat{\boldsymbol{w}}_{k|\boldsymbol{\Phi}_\ell}(i)$ is the least mean-square estimate of $\boldsymbol{w}_k^o$ given $\boldsymbol{\Phi}_\ell(i)$. A larger $\xi_{\rm net}(i)$ implies better average privacy preservation across the network at time instant i.

In our numerical experiments, we compare the performance of our proposed ATP(δ) algorithm with the proposed ATP(0) algorithm, the multitask diffusion algorithm [7] (denoted as MDA for convenience) and the non-cooperative LMS algorithm (24) (denoted as NoCoop). Since the unknown coefficient vectors $\{\boldsymbol{w}_k^o\}$ are linearly related across the network, agent ℓ can make use of all neighboring estimates $\{\boldsymbol{\psi}_s'(i)\}_{s\in\mathcal{N}_\ell}$ that are available to it at time instant i to infer its neighbor's task $\boldsymbol{w}_k^o$ for any $k\in\mathcal{N}_\ell$. However, for simplicity, in order to evaluate the privacy-preserving performance of the different schemes, we constrain the information set $\boldsymbol{\Phi}_\ell(i)$ as follows:

- For ATP(δ), we let $\boldsymbol{\Phi}_\ell(i)=\{\boldsymbol{\psi}_\ell(i),\boldsymbol{\psi}_k'(i)\}$, i.e., each agent ℓ is memoryless across time and spatial domains.
- For ATP(0) and MDA, we let $\boldsymbol{\Phi}_\ell(i)=\{\boldsymbol{\psi}_\ell(i),\boldsymbol{\psi}_k(i)\}$ since no privacy noises are added in these schemes.
- For NoCoop, we let $\boldsymbol{\Phi}_\ell(i)=\{\boldsymbol{w}_\ell(i)\}$ since there is no information exchange between agents.

Let

$$\mathcal{W}=\mathbb{E}[(\boldsymbol{w}^o-\mathbb{E}\boldsymbol{w}^o)(\boldsymbol{w}^o-\mathbb{E}\boldsymbol{w}^o)^{\mathsf{T}}]\in\mathbb{R}^{M\times M}, \tag{81}$$

$$\Psi(i)=\mathbb{E}[\boldsymbol{\psi}(i)\boldsymbol{\psi}^{\mathsf{T}}(i)]\in\mathbb{R}^{M\times M}, \tag{82}$$

$$\mathcal{V}(i)=\mathbb{E}\Big[(\boldsymbol{w}^o-\mathbb{E}\boldsymbol{w}^o)(\widetilde{\boldsymbol{\psi}}(i)-\mathbb{E}\widetilde{\boldsymbol{\psi}}(i))^{\mathsf{T}}\Big]\in\mathbb{R}^{M\times M}, \tag{83}$$

$$\boldsymbol{r}_{du}(i)=\mathrm{col}\{\boldsymbol{u}_k(i)\boldsymbol{d}_k(i)\}_{k=1}^N=\boldsymbol{\mathcal{R}}_u(i)\boldsymbol{w}^o+\boldsymbol{g}(i)\in\mathbb{R}^{M\times 1}, \tag{84}$$

$$\mathcal{B}(i)=(I_M-\mathcal{M}\mathcal{R}_u)\mathcal{P}(i)\in\mathbb{R}^{M\times M}. \tag{85}$$

Let $\boldsymbol{\psi}_{\{k,\ell\}}(i)=[\boldsymbol{\psi}_k^{\mathsf{T}}(i),\boldsymbol{\psi}_\ell^{\mathsf{T}}(i)]^{\mathsf{T}}\in\mathbb{R}^{M_{\{k,\ell\}}\times 1}$, where $M_{\{k,\ell\}}\triangleq M_k+M_\ell$. We also define

$$\begin{aligned}U_{k\{k,\ell\}}(i)=\mathbb{E}\Big[&(\boldsymbol{w}_k^o-\mathbb{E}\boldsymbol{w}_k^o)\\&\times\left(\boldsymbol{\psi}_{\{k,\ell\}}(i)-\mathbb{E}\boldsymbol{\psi}_{\{k,\ell\}}(i)\right)^{\mathsf{T}}\Big]\in\mathbb{R}^{M_k\times M_{\{k,\ell\}}},\end{aligned} \tag{86}$$

$$\begin{aligned}X_{\{k,\ell\}\{k,\ell\}}(i)=\mathbb{E}\Big[&\left(\boldsymbol{\psi}_{\{k,\ell\}}(i)-\mathbb{E}\boldsymbol{\psi}_{\{k,\ell\}}(i)\right)\\&\times\left(\boldsymbol{\psi}_{\{k,\ell\}}(i)-\mathbb{E}\boldsymbol{\psi}_{\{k,\ell\}}(i)\right)^{\mathsf{T}}\Big]\in\mathbb{R}^{M_{\{k,\ell\}}\times M_{\{k,\ell\}}},\end{aligned} \tag{87}$$

and

$$\breve{R}_{n,k}(i)=\begin{bmatrix}R_{n,k}(i)&\mathbf{0}_{M_k\times M_\ell}\\\mathbf{0}_{M_\ell\times M_k}&\mathbf{0}_{M_\ell\times M_\ell}\end{bmatrix}$$

where the quantity $R_{n,k}(i)$ is defined by (13). Let

$$\begin{aligned}\mathcal{H}(i)=\mathbb{E}\,[\,&\left(\left(I_M-\mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)\right)\mathcal{P}(i)\right)\\&\otimes\left(\left(I_M-\mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)\right)\mathcal{P}(i)\right)\,]\in\mathbb{R}^{M^2\times M^2},\end{aligned} \tag{88}$$

$$\begin{aligned}\mathcal{X}(i)=\mathbb{E}\,[\,&\left(\left(I_M-\mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)\right)f(i)\right)\\&\otimes\left(\left(I_M-\mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)\right)\mathcal{P}(i)\right)\,]\in\mathbb{R}^{M^2\times M},\end{aligned} \tag{89}$$

$$\begin{aligned}\mathcal{X}'(i)=\mathbb{E}\,[\,&\left(\left(I_M-\mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)\right)\mathcal{P}(i)\right)\\&\otimes\left(\left(I_M-\mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)\right)f(i)\right)\,]\in\mathbb{R}^{M^2\times M},\end{aligned} \tag{90}$$

$$\begin{aligned}\mathcal{Y}(i)=\mathbb{E}\,[\,&\left(\mathcal{M}\boldsymbol{r}_{du}(i+1)\right)\\&\otimes\left(\left(I_M-\mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)\right)\mathcal{P}(i)\right)\,]\in\mathbb{R}^{M^2\times M},\end{aligned} \tag{91}$$

$$\mathcal{Y}'(i) = \mathbb{E}\,[\,((I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1))\,\mathcal{P}(i)) \otimes (\mathcal{M}\boldsymbol{r}_{du}(i+1))\,] \in \mathbb{R}^{M^2\times M}. \quad (92)$$

We also define the following $M \times M$ matrices:

$$\begin{aligned} C_1(i) &= \mathbb{E}\,[\,(I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1))\,f(i) \times f^{\mathsf{T}}(i)\,(I_M - \boldsymbol{\mathcal{R}}_u(i+1)\mathcal{M})\,], \\ C_2(i) &= \mathbb{E}\,[\,(I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1))\,f(i)\boldsymbol{r}_{du}^{\mathsf{T}}(i+1)\,]\mathcal{M}, \\ C_3(i) &= \mathbb{E}\,[\,\mathcal{M}\boldsymbol{r}_{du}(i+1)f^{\mathsf{T}}(i)\,(I_M - \boldsymbol{\mathcal{R}}_u(i+1)\mathcal{M})\,], \\ C_4 &= \mathcal{M}\mathbb{E}\,[\,\boldsymbol{r}_{du}(i+1)\boldsymbol{r}_{du}^{\mathsf{T}}(i+1)\,]\mathcal{M}, \end{aligned}$$

and the following $M^2 \times 1$ vector:

$$c(i) = \mathrm{vec}\,(C_1(i)) - \mathrm{vec}\,(C_2(i)) - \mathrm{vec}\,(C_3(i)) + \mathrm{vec}\,(C_4)\,. \quad (93)$$

We notice that all the variables defined by (88) through (93) except C_4 depend on the quantities $\{\mathcal{P}(i), f(i)\}$ at time instant i. We also define the following $M^2 \times M^2$ constant matrix:

$$\mathcal{Z} = \mathbb{E}[((I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1))) \otimes ((I_M - \mathcal{M}\boldsymbol{\mathcal{R}}_u(i+1)))]. \quad (94)$$

In the following Theorem 4, we evaluate the privacy-preserving performance of the proposed ATP(δ) algorithm.

Theorem 4. *Consider the distributed strategy (9) to (12). Suppose that Assumptions 1 to 4 hold. The network inference error at each time instant $i \geq 0$ is*

$$\xi_{net}^{ATP(\delta)}(i) = \frac{1}{N}\sum_{k=1}^{N}\frac{1}{|\mathcal{N}_k|-1}\sum_{\ell\in\mathcal{N}_k\backslash\{k\}}\mathrm{Tr}\Big(W_{kk} - U_{k\{k,\ell\}}(i)\Big(X_{\{k,\ell\}\{k,\ell\}}(i) + \breve{R}_{n,k}(i)\Big)^{-1}U_{k\{k,\ell\}}^{\mathsf{T}}(i)\Big) \quad (95)$$

where

$$U_{k\{k,\ell\}}(i) = \Big[[\mathcal{W} - \mathcal{V}(i)]_{k,k}, [\mathcal{W} - \mathcal{V}(i)]_{k,\ell}\Big], \quad (96)$$

$$\mathcal{V}(i+1) = \mathcal{V}(i)\mathcal{B}^{\mathsf{T}}(i), \quad (97)$$

$$\mathcal{V}(0) = \mathcal{W}\,(I_M - \mathcal{M}\mathcal{R}_u), \quad (98)$$

and where

$$X_{\{k,\ell\}\{k,\ell\}}(i) = \begin{bmatrix} [\Psi(i) - \mathbb{E}\boldsymbol{\psi}(i)(\mathbb{E}\boldsymbol{\psi}(i))^{\mathsf{T}}]_{k,k} & [\Psi(i) - \mathbb{E}\boldsymbol{\psi}(i)(\mathbb{E}\boldsymbol{\psi}(i))^{\mathsf{T}}]_{k,\ell} \\ [\Psi(i) - \mathbb{E}\boldsymbol{\psi}(i)(\mathbb{E}\boldsymbol{\psi}(i))^{\mathsf{T}}]_{\ell,k} & [\Psi(i) - \mathbb{E}\boldsymbol{\psi}(i)(\mathbb{E}\boldsymbol{\psi}(i))^{\mathsf{T}}]_{\ell,\ell} \end{bmatrix} \quad (99)$$

with

$$\mathbb{E}\boldsymbol{\psi}(0) = \mathcal{M}\mathcal{R}_u\mathbb{E}\boldsymbol{w}^o, \quad (100)$$

$$\mathbb{E}\boldsymbol{\psi}(i+1) = (I_M - \mathcal{M}\mathcal{R}_u)\,\mathcal{P}(i)\mathbb{E}\boldsymbol{\psi}(i) - (I_M - \mathcal{M}\mathcal{R}_u)\,f(i) + \mathcal{M}\mathcal{R}_u\mathbb{E}\boldsymbol{w}^o, \quad (101)$$

and

$$\Psi(0) = \mathcal{M}\mathbb{E}[\boldsymbol{\mathcal{R}}_u(0)\,\mathbb{E}[\boldsymbol{w}^o\,(\boldsymbol{w}^o)^{\mathsf{T}}]\boldsymbol{\mathcal{R}}_u(0)]\mathcal{M} + \mathcal{M}\mathcal{G}\mathcal{M}, \quad (102)$$

$$\mathrm{vec}\,(\Psi(i+1)) = \mathcal{H}(i)\,\mathrm{vec}\,(\Psi(i)) + \mathcal{Z}\,\mathrm{vec}(\Gamma(i)) + c(i) + (\mathcal{Y}(i) - \mathcal{X}(i) - \mathcal{X}'(i) + \mathcal{Y}'(i))\,\mathbb{E}\boldsymbol{\psi}(i). \quad (103)$$

Proof. See Section S2 of the supplementary material. □

VI. Simulation Results

In this section, we compare performance of the proposed ATP(δ) algorithm in terms of network inference privacy and network MSD against ATP(0), MDA [7] and NoCoop (24) in two types of networks: 1) a line network with a large number of linear equality constraints; and 2) a very dense network with a small number of linear equality constraints. We also test the tracking performance of the proposed distributed and adaptive ATP(δ) algorithm in the case of changing statistics. In the last example, we utilize the proposed ATP(δ) algorithm to solve a statistical disclosure limitation problem.

A. Selection of Weights for Weighted Projection

Note that the projection step (12) at each agent k involves noisy intermediate estimates $\{\boldsymbol{\psi}'_\ell(i)\}$ from its neighbors $\{\ell \in \mathcal{N}_k\backslash\{k\}\}$ and the uncontaminated intermediate estimate $\boldsymbol{\psi}_k(i)$. In view of the fact that the noisy intermediate estimate $\boldsymbol{\psi}'_\ell(i)$ involves a privacy noise $\boldsymbol{n}_\ell(i)$ with variance $\sigma^2_{n,\ell}(i)$, we set the weights $\{\theta_{\ell k}(i)\}$ defined by (15) to

$$\theta_{\ell k}(i) = \begin{cases} \dfrac{e^{-\sigma^2_{n,\ell}(i)}}{\sum_{\ell\in\mathcal{N}_k\backslash\{k\}} e^{-\sigma^2_{n,\ell}(i)}+1}, & \text{if } \ell \in \mathcal{N}_k\backslash\{k\} \\ \dfrac{1}{\sum_{\ell\in\mathcal{N}_k\backslash\{k\}} e^{-\sigma^2_{n,\ell}(i)}+1}, & \text{if } \ell = k \end{cases} \quad (104)$$

which satisfies the requirements in (16). We observe from (104) the following: 1) For any agent k and time instant $i \geq 0$, the largest weight among the weights $\{\theta_{\ell k}(i)\}_{\ell\in\mathcal{N}_k}$ is $\theta_{kk}(i)$, which is reasonable because the intermediate estimate $\boldsymbol{\psi}_k(i)$ is uncontaminated. 2) At each agent k and time instant $i \geq 0$, a small value is assigned to the weight $\theta_{\ell k}(i)$ for its neighboring agent $\ell \in \mathcal{N}_k\backslash\{k\}$ with a large privacy noise power $\sigma^2_{n,\ell}(i)$, with $\theta_{\ell k}(i) \to 0$ as $\sigma^2_{n,\ell}(i) \to \infty$.

B. Line Network

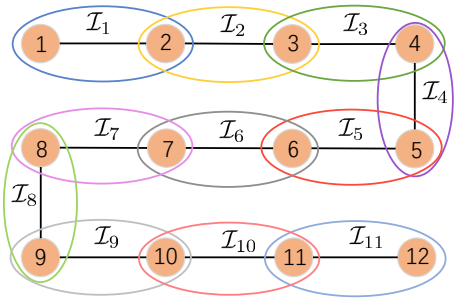
As shown in Fig. 2a, we consider a line network which consists of $N = 12$ agents. Recalling the assumptions about (5), we notice that all agents in each constraint set $\mathcal{I}_q$ are required to be neighboring agents to allow a distributed processing of the q-th constraint, for any $q = 1, \ldots, Q$. Considering the line network in Fig. 2a, we set up $Q = 11$ total linear equality constraints in the network, where each $\mathcal{I}_q$ for any $q = 1, \ldots, Q$ only contains two neighboring agents that are connected by an edge. Each linear equality constraint is of the form [7]:

$$\sum_{k\in\mathcal{I}_q} d_{qk}\boldsymbol{w}_k^o + b_q\mathbf{1}_{3\times 1} = \mathbf{0}_{3\times 1}, \quad q = 1, \ldots, Q \quad (105)$$

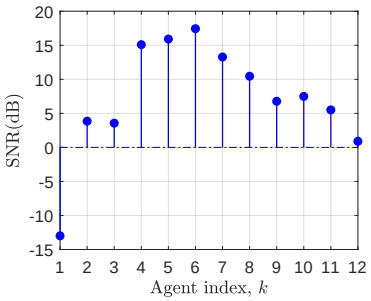
with the scalar parameters $\{d_{qk}, b_q\}$ randomly selected from $[-3,-1] \cup [1,3]$. The lengths of the unknown parameter vectors $\{\boldsymbol{w}_k^o\}$ are identical across the agents, and we set $M_k = 3$ for $k = 1, \ldots, N$. The random data $\{\boldsymbol{u}_k(i), \boldsymbol{v}_k(i)\}$ are independent, normally distributed with zero mean, and white over time and space. Let

$$\mathrm{SNR}_k = 10\log_{10}\left(\mathbb{E}\big[(\boldsymbol{u}_k^{\mathsf{T}}(i)\boldsymbol{w}_k^o)^2\big]/\sigma^2_{v,k}\right)$$

be the signal-to-noise ratio (SNR) at agent k. Then, the parameters $\{R_{u,k}, W_{kk}, \mathbb{E}\boldsymbol{w}_k^o, \sigma^2_{v,k}\}$ are adjusted such that the SNR values $\{\mathrm{SNR}_k : k = 1, \ldots, N\}$ are as shown in



(a)



(b)

Fig. 2. (a) A line network and (b) SNRs across the agents in a line network.

Fig. 2b. For the step-size parameters, we set $\mu_k = 0.02$ for all $k = 1, \ldots, N$ in all the tested algorithms except MDA, where we let $\mu_k/j_k = 0.02$ instead to ensure the same step-size in the adaptation step for all the above-mentioned algorithms. Let $\delta_k \triangleq \rho \operatorname{Tr}(W_{kk})$ with $0 \le \rho \le 1$. Then, for the ATP(δ) algorithm, we set $\rho = 0.1$, $\rho = 0.6$, $\rho = 0.85$, respectively, to compare performance under different privacy thresholds $\{\delta_k\}$. In addition, for all k, we set the privacy noise power $\sigma^2_{n,k}(i) = \widehat{\sigma}^2_{n,k}(i)$ defined in (63).

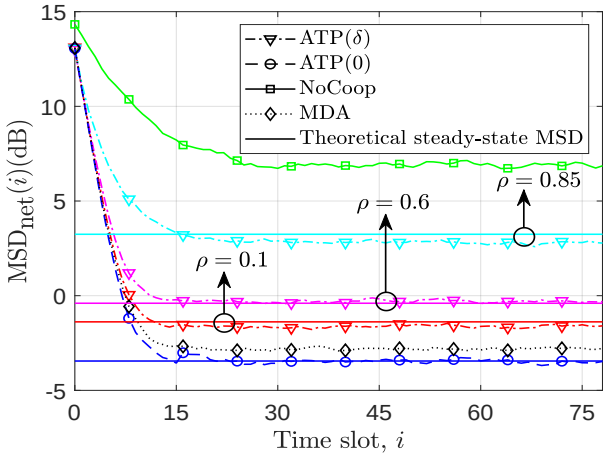


Fig. 3. Network MSD learning curves of MDA [7], NoCoop, the proposed ATP(δ) under different parameter settings for δ, and ATP(0) in a line network.

Fig. 3 shows the network MSD learning curves of the tested strategies, which are averaged over 1,000 independent realizations of $\{\boldsymbol{w}_k^o\}$. We observe that 1) simulation results match well with theoretical findings, 2) by adding privacy noises, the proposed ATP(δ) algorithm converges to a higher MSD level than ATP(0) and MDA [7], 3) larger privacy thresholds result in a higher steady-state network MSD. These results match well with the theoretical findings in Lemma 2. Fig. 4 plots the network inference error, defined by (80). We observe that 1) the network inference privacy performance of ATP(0) and MDA [7] are similar, but worse (lower values of $\{\xi_{\mathrm{net}}(i)\}$) than the proposed ATP($\delta$), 2) by increasing privacy thresholds, the proposed ATP(δ) algorithm shows better network inference privacy performance. These results demonstrate that the proposed ATP(δ) algorithm is able to balance the trade-off between network MSD and privacy protection. Note that the proposed ATP(δ) algorithm aims at protecting each agent's

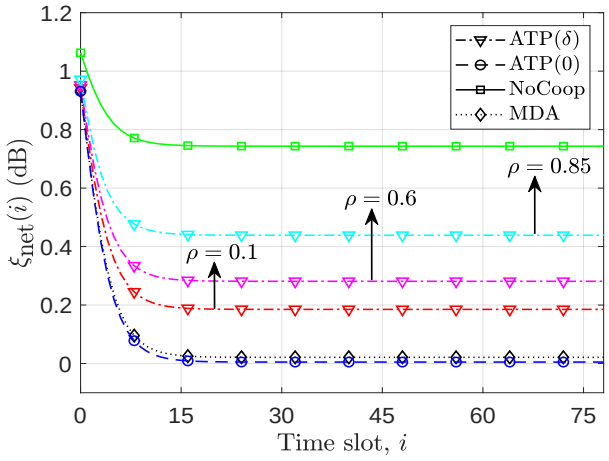


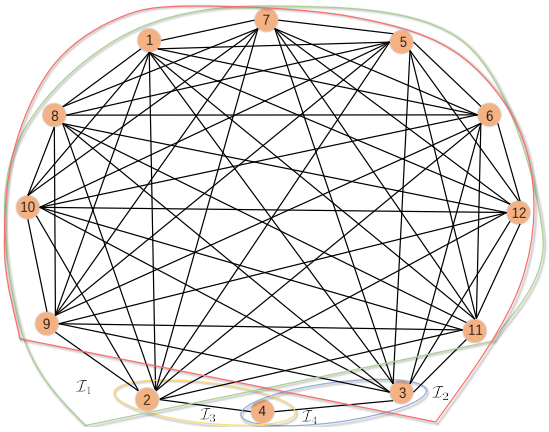
Fig. 4. Network inference privacy learning curves of MDA [7], NoCoop, the proposed ATP(δ) under different parameter settings for δ, and ATP(0) in a line network.

local task by adding a privacy noise to its local estimate before sharing with its neighbors, which results in a larger steady-state network MSD in view of Lemma 2. Then, by adjusting the values of the privacy thresholds, the ATP(δ) algorithm is able to satisfy the requirements of different trade-offs between network MSD and privacy protection.

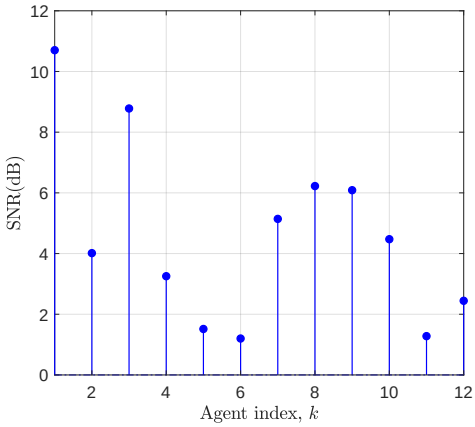
C. Dense Network

As shown in Fig. 5a, we consider a dense network which consists of $N = 12$ agents. The agents in the network are involved in $Q = 4$ linear equality constraints, each of the form (105). We observe from Fig. 5a that all agents in each constraint set $\mathcal{I}_q$, for any $q = 1, \ldots, 4$, are neighboring agents, which allows a distributed processing of each agent's local constraints. The length of the unknown parameter vector $\boldsymbol{w}_k^o$ is $M_k = 3$ for every k. The random data $\{\boldsymbol{u}_k(i), \boldsymbol{v}_k(i)\}$ are independent, normally distributed with zero mean, and white over time and space. The parameters $\{R_{u,k}, W_{kk}, \mathbb{E}\boldsymbol{w}_k^o, \sigma^2_{v,k}\}$ are adjusted such that the SNR values $\{\mathrm{SNR}_k\}$ are as shown in Fig. 5b. For the ATP(δ) algorithm, we set the threshold values $\{\delta_k = 0.1 \operatorname{Tr}(W_{kk})\}_{k=1}^N$. The other parameter settings remain the same as those in the previous simulation.

Fig. 6 shows the network MSD learning curves of the tested strategies, which are averaged over 20,000 independent realizations of $\{\boldsymbol{w}_k^o\}$. Fig. 7 plots the network inference



(a)



(b)

Fig. 5. (a) A dense network and (b) SNRs across the agents in a dense network.

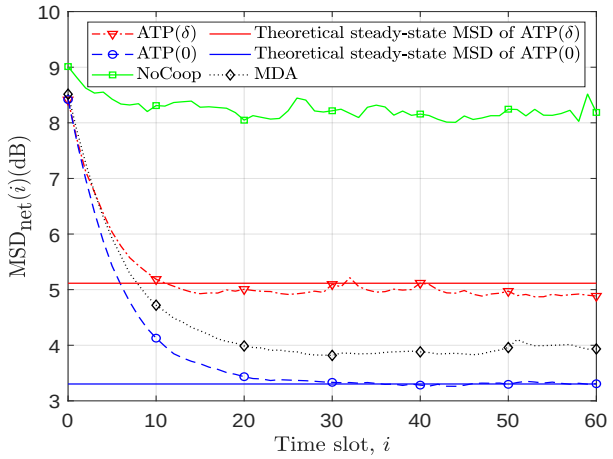


Fig. 6. Network MSD learning curves of MDA [7], NoCoop, the proposed ATP(δ) and ATP(0) in a dense network.

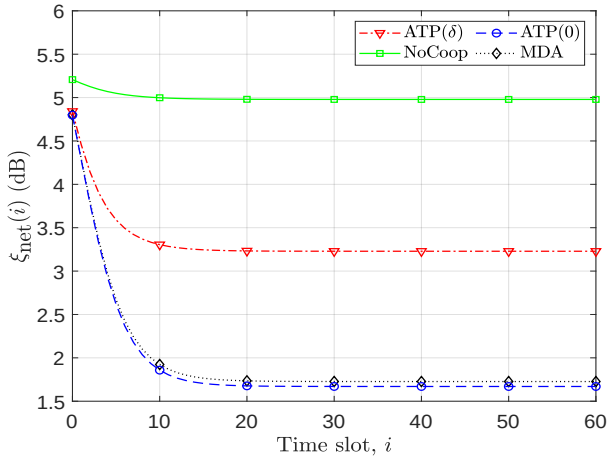


Fig. 7. Network inference privacy learning curves of MDA [7], NoCoop, the proposed ATP(δ) and ATP(0) in a dense network.

error, defined by (80). These results demonstrate that the proposed ATP(δ) algorithm is still able to balance the trade-off between network MSD and privacy protection in dense networks. Table I lists the ratios of privacy gain to accuracy loss in the line and dense network, where the privacy gain is the absolute gap between the steady-state network inference privacy errors of ATP(0) and ATP(δ) in dB, and the accuracy loss is the absolute gap between the steady-state network MSDs in dB. We observe that the ratio in the dense network is much higher than in the line network, which means that more privacy gains are obtained in the former case with the same quantity of network MSD loss. These results demonstrate that the proposed ATP(δ) algorithm performs better in dense networks than in line networks.

TABLE I
RATIO OF PRIVACY GAIN TO ACCURACY LOSS

Line Network			Dense Network
$\rho = 0.1$	$\rho = 0.6$	$\rho = 0.85$	
0.09	0.09	0.06	0.86

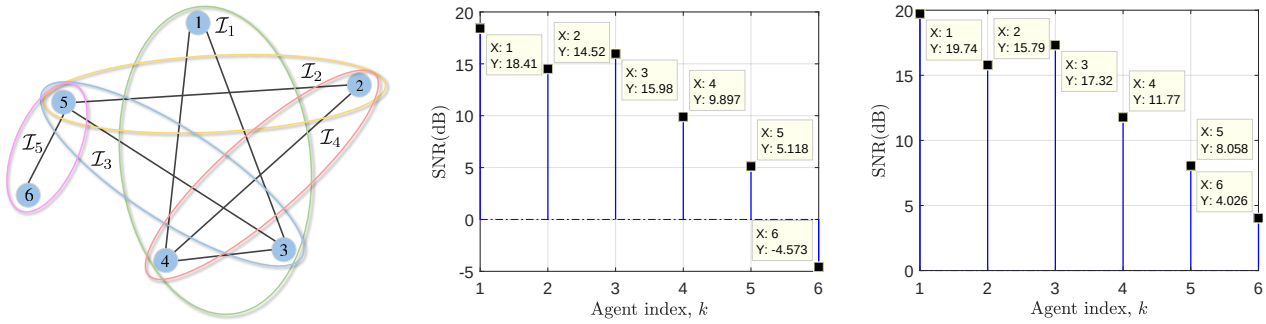
D. Tracking Performance

As shown in Fig. 8a, we consider the case where there are $N = 6$ agents in the network. The agents in the network are involved in $Q = 5$ linear equality constraints, each of the form [7]:

$$\sum_{k\in\mathcal{I}_q} d_{qk}\boldsymbol{w}_k^o + b_q\mathbf{1}_{2\times 1} = \mathbf{0}_{2\times 1}, \quad q = 1,\ldots,Q$$

with the scalar parameters $\{d_{qk}, b_q\}$ randomly selected from $[-3,-1]\cup[1,3]$. The lengths of the unknown parameter vectors $\{\boldsymbol{w}_k^o\}$ are $M_k = 2$ for all $k = 1,\ldots,N$.

The random data $\{\boldsymbol{u}_k(i), \boldsymbol{v}_k(i)\}$ are independent, normally distributed with zero mean, and white over time and space. The parameters $\{R_{u,k}, W_{kk}, \mathbb{E}\boldsymbol{w}_k^o, \sigma_{v,k}^2\}$ are adjusted such that the SNR values $\{\text{SNR}_k\}$ are as shown in Fig. 8b at the first stage. Then in order to test the tracking performance of the proposed distributed and adaptive ATP(δ) algorithm, we increase values of the elements in W_{kk} at time instant $i = 75$, which results in the $\{\text{SNR}_k\}$ as shown in Fig. 8c at the second stage. For the step-size parameters, we set $\mu_k = 0.02$ for all $k = 1,\ldots,N$ in all the tested algorithms except MDA, where we let $\mu_k/j_k = 0.02$ to ensure the same



(a) (b) (c)

Fig. 8. (a) Network topology, (b) SNRs across the agents at the first stage and (c) SNRs across the agents at the second stage.

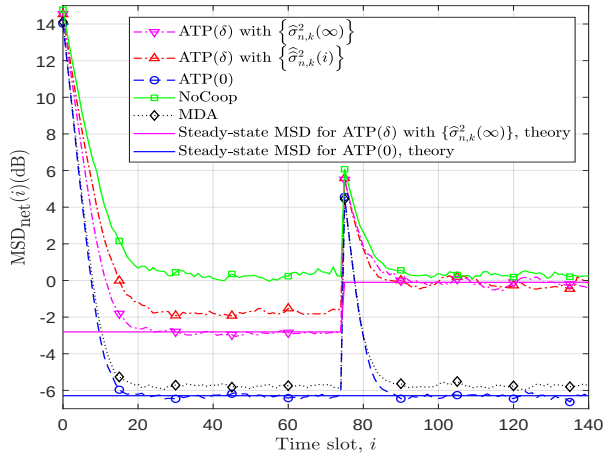


Fig. 9. Network MSD learning curves of MDA [7], NoCoop, the proposed ATP(0), and ATP(δ) with privacy noise powers $\left\{\sigma^2_{n,k}(i)=\widehat{\sigma}^2_{n,k}(\infty)\right\}$ and $\left\{\sigma^2_{n,k}(i)=\widehat{\widehat{\boldsymbol{\sigma}}}^2_{n,k}(i)\right\}$, with the quantities $\widehat{\sigma}^2_{n,k}(\infty)$ and $\widehat{\widehat{\boldsymbol{\sigma}}}^2_{n,k}(i)$ defined in (68) and (69), respectively.

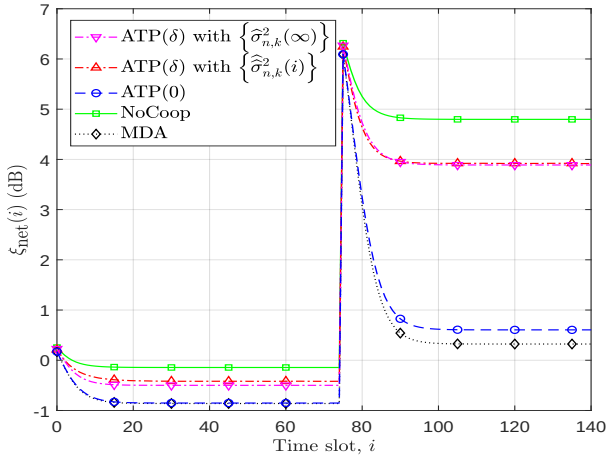


Fig. 10. Network inference privacy learning curves of MDA [7], NoCoop, the proposed ATP(0), and ATP(δ) with privacy noise powers $\left\{\sigma^2_{n,k}(i)=\widehat{\sigma}^2_{n,k}(\infty)\right\}$ and $\left\{\sigma^2_{n,k}(i)=\widehat{\widehat{\boldsymbol{\sigma}}}^2_{n,k}(i)\right\}$, with the quantities $\widehat{\sigma}^2_{n,k}(\infty)$ and $\widehat{\widehat{\boldsymbol{\sigma}}}^2_{n,k}(i)$ defined in (68) and (69), respectively.

step-size in the adaptation step for all the above-mentioned algorithms. In addition, for the ATP(δ) algorithm, we set the threshold values $\{\delta_k = 0.6\ \mathrm{Tr}(W_{kk})\}_{k=1}^N$ according to the covariance matrix W_{kk} at the first stage. We test both cases where we set the privacy noise powers $\left\{\sigma^2_{n,k}(i)=\widehat{\sigma}^2_{n,k}(\infty)\right\}$ and $\left\{\sigma^2_{n,k}(i)=\widehat{\widehat{\boldsymbol{\sigma}}}^2_{n,k}(i)\right\}$, with the quantities $\widehat{\sigma}^2_{n,k}(\infty)$ and $\widehat{\widehat{\boldsymbol{\sigma}}}^2_{n,k}(i)$ defined in (68) and (69), respectively.

Fig. 9 shows the network MSD learning curves of the tested strategies, which are averaged over 1,000 independent runs. We observe that by increasing values of the elements in W_{kk}, the proposed ATP(δ) algorithm with privacy noise powers $\left\{\sigma^2_{n,k}(i)=\widehat{\sigma}^2_{n,k}(\infty)\right\}$ converges to a higher MSD level at the second stage. This is because privacy noise powers become larger at the second stage according to (68). We also note that the steady-state network MSD of the proposed distributed and adaptive ATP(δ) gets higher at the second stage. Fig. 10 plots the network inference error, defined by (80). We observe that the network inference privacy performance of both ATP(δ) algorithms are similar throughout the whole process. These results demonstrate that the proposed distributed and adaptive ATP(δ) algorithm is able to track changes of the covariance matrix W_{kk}.

E. Statistical Disclosure Limitation

The statistical disclosure limitation problem is to choose a methodology for data release so that disclosure risk is adequately low while statistical information (data utility) in the disseminated data is as high as possible [61]–[63]. Consider Table II with 9 internal cells and 6 marginal cells and one overall total cell value, where the cell values in thousand tonnes of oil equivalent (ktoe) are from consumption statistics for fuels used in road transport in 2019 released by the Office of National Statistics in the U.K.. The objective is to modify the cell values in Table II for data release in order to reduce the disclosure risk of the internal cell values while the marginal cell values remain largely unchanged.

We start by representing Table II as a flow network with source s, sink t, and 6 additional nodes as shown in Fig. 11. Specifically, we associate each row and each column of Table II with a node, and each cell with an oriented arc. We let each cell value denote the amount of flow on an arc. Let $\{c_j\}_{j=1}^9$ denote the 9 internal cell values in Table II, and $\{m_k\}_{k=1}^6$ the marginal cell values. Each marginal cell value m_k, for $k=1,\ldots,6$, is the sum of three internal cell values $\{c_j\}$, leading to the flow conservation condition that for any

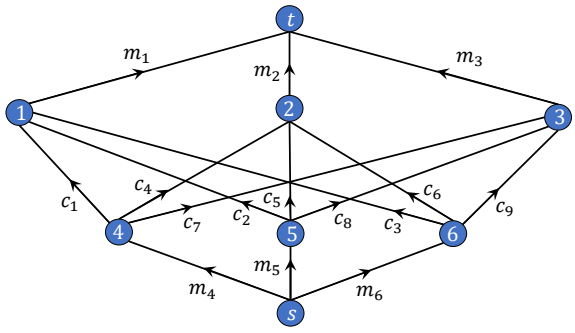


Fig. 11. Flow network topology with source s, sink t, and 6 additional nodes, where nodes $1, 2, \ldots, 6$ correspond to Barnsley, Calderdale, Harrogate, Motorways, A roads and Minor roads in Table II, respectively.

node except for s and t, the flow in to the node must equal the flow out from the node in Fig. 11.

We assume that the published internal cell values $\{c_j\}_{j=1}^9$ in Table II are modified data using some disclosure limitation method, and the original value $\boldsymbol{c}_j^o$, for any $j = 1, \ldots, 9$, is a random variable with mean c_j and variance $\sigma_{c,j}^2$. We note that $\boldsymbol{c}_j^o > 0$ for any $j = 1, \ldots, 9$. Then, we associate each node k with a 3×1 unknown parameter vector of interest $\boldsymbol{w}_k^o$, which consists of the amounts of flows on the arcs between node k and three other nodes. For instance, we have

$$\boldsymbol{w}_1^o = [\boldsymbol{c}_1^o, \boldsymbol{c}_2^o, \boldsymbol{c}_3^o]^\mathsf{T} \text{ and } \boldsymbol{w}_5^o = [\boldsymbol{c}_2^o, \boldsymbol{c}_5^o, \boldsymbol{c}_8^o]^\mathsf{T}.$$

In order to limit the disclosure risk of internal cell values, we assume that the unknown parameter vector $\boldsymbol{w}_k^o$ is private for each node k. At each time instant $i \geq 0$, each node k in the network has access to a noisy measurement $\boldsymbol{m}_k(i)$ of the amount of flow on the arc between node k and s or t. It then follows from the flow conservation condition that

$$\boldsymbol{m}_k(i) = \mathbf{1}_{3\times 1}^\mathsf{T} \boldsymbol{w}_k^o + \boldsymbol{v}_k(i),$$

where the scalar $\boldsymbol{v}_k(i)$ denotes a random measurement noise at time instant i, with zero mean and variance $\sigma_{v,k}^2$. Each node k in the network seeks to cooperate with its neighbors to estimate the unknown parameter vector of interest $\boldsymbol{w}_k^o$ from noisy measurements $\boldsymbol{m}_k(i)$, as well as to protect its individual task against privacy leakage. In order to reduce the information loss of marginal cell values as much as possible, we proceed to solve the following optimization problem:

$$\min_{w_1,\ldots,w_6} \sum_{k=1}^{6} \mathbb{E}\Big[\big(\boldsymbol{m}_k(i) - \mathbf{1}_{3\times 1}^\mathsf{T} w_k\big)^2 \,\Big|\, \boldsymbol{w}_k^o = w_k^o\Big] \tag{106a}$$

$$\text{s. t. } [w_k]_{(\ell,k)} + [w_\ell]_{(k,\ell)} = 0, \ \ell \in \mathcal{N}_k, \text{ for all } k, \tag{106b}$$

where the notation $[w_k]_{(\ell,k)}$ denotes the amount of flow on the arc from node ℓ to node k, and returns a positive value for a flow into node k and a negative one for a flow out of node k.

In our experiment, each internal cell value $\boldsymbol{c}_j^o$ for any $j = 1, \ldots, 9$ is normally distributed with mean c_j and variance $\sigma_{c,j}^2 = 5 \times 10^{-4} c_j$. The measurement noise $\boldsymbol{v}_k(i)$ follows the normal distribution with zero mean and variance $\sigma_{v,k}^2 = 0.01 m_k$ for each $k = 1, \ldots, 6$. We now proceed to utilize the proposed ATP(δ) and ATP(0) algorithms to solve the optimization problem (106). We set the step-size $\mu_k = 6 \times 10^{-3}$ for all $k = 1, \ldots, 6$ in both algorithms. In the proposed ATP(δ) algorithm, we set the privacy thresholds $\{\delta_k = 0.1 \operatorname{Tr}(W_{kk})\}$, and the privacy noise powers $\left\{\sigma_{n,k}^2(i) = \widehat{\sigma}_{n,k}^2(\infty)\right\}$ with the quantity $\widehat{\sigma}_{n,k}^2(\infty)$ defined in (68). Tables III and IV list the average relative errors of the estimated cell values using the proposed ATP(0) and ATP(δ) algorithm, respectively, w.r.t. the original cell values in Table II over 100 independent realizations of $\{\boldsymbol{c}_j^o\}$. Given some value $v \neq 0$ and its approximation $\widehat{v}$, the relative error is

$$\eta = \frac{|v - \widehat{v}|}{|v|}.$$

We note that 1) small relative errors w.r.t. the marginal cell values in Table II indicate low information loss of marginal cell values, 2) large relative errors w.r.t. the internal cell values in Table II indicate low disclosure risk of internal cell values. In order to compare the information loss and disclosure risk of the proposed ATP(δ) and ATP(0) algorithms, we compute the ratios of the relative errors in Table IV to those in Table III and find that 1) in terms of the relative errors w.r.t. the internal cell values, the average ratio is 2.185, which means that the corresponding relative errors of the ATP(δ) algorithm are, on average, twice as large as those of the ATP(0) algorithm, 2) in terms of the relative errors w.r.t. the marginal cell values, the average ratio is 1.049, which means that the corresponding relative errors of both algorithms are almost the same. These results demonstrate that by adding privacy noises, the proposed ATP(δ) algorithm is able to reduce the disclosure risk of the internal cell values while the information loss of the marginal cell values remains largely unchanged.

TABLE II
ROAD TRANSPORT ENERGY CONSUMPTION IN KTOE AT LOCAL AUTHORITY LEVEL, 2019

	Motorways	A roads	Minor roads	Total
Barnsley	$c_1 = 13,295$	$c_2 = 12,986$	$c_3 = 19,398$	$m_1 = 45,679$
Calderdale	$c_4 = 12,145$	$c_5 = 10,675$	$c_6 = 13,945$	$m_2 = 36,765$
Harrogate	$c_7 = 18,852$	$c_8 = 13,170$	$c_9 = 12,764$	$m_3 = 44,786$
Total	$m_4 = 44,292$	$m_5 = 36,831$	$m_6 = 46,107$	$127,230$

TABLE III
RELATIVE ERRORS OF ATP(0) ALGORITHM

	Motorways	A roads	Minor roads	Total ($\times 10^{-5}$)
Barnsley	0.1925	0.0293	0.1515	9.32
Calderdale	0.0607	0.0262	0.0329	10.08
Harrogate	0.1748	0.0077	0.2662	8.74
Total ($\times 10^{-5}$)	9.45	9.89	9.65	5.54

VII. CONCLUSION

We have developed a privacy-preserving distributed strategy over linear multitask networks, which is able to protect each agent's local task by adding a privacy noise to its local information before sharing with its neighbors. We proposed a

TABLE IV
RELATIVE ERRORS OF ATP(δ) ALGORITHM

	Motorways	A roads	Minor roads	Total ($\times 10^{-5}$)
Barnsley	0.2191	0.0367	0.1257	9.59
Calderdale	0.0245	0.1089	0.1047	9.42
Harrogate	0.1602	0.0514	0.2898	9.44
Total ($\times 10^{-5}$)	11.07	10.38	11.43	4.94

utility-privacy optimization trade-off to determine the amount of noise to add, and derived a sufficient condition for the privacy noise powers in order to satisfy the proposed privacy constraints. Furthermore, we proposed a distributed and adaptive scheme to compute the privacy noise powers. We have studied the mean and mean-square behaviors and privacy-preserving performance. Simulation results demonstrated that the proposed scheme is able to balance the trade-off between the network MSD and network inference privacy.

Future work concerns privacy issues over multitask networks where agents' local parameters of interest consist of common parameters within each neighborhood and individual parameters. Privacy-preserving distributed strategies will be developed to improve the estimation accuracy of common parameters at each agent via in-network cooperation with neighboring agents, as well as to protect individual parameters against privacy leakage.

APPENDIX A
PROOF OF THEOREM 1

The $M_k \times M_k$ symmetric positive semi-definite matrix $X_{kk}(i)$ admits a spectral decomposition of the form:

$$X_{kk}(i) = T_k(i)\Lambda_k(i)T_k^{\intercal}(i),$$

where $T_k(i)$ is an orthogonal matrix and $\Lambda_k(i) = \operatorname{diag}\{\lambda_k^m(i)\}_{m=1}^{M_k}$ is a diagonal matrix consisting of the eigenvalues of $X_{kk}(i)$. Let

$$\widetilde{U}_{kk}(i) = T_k^{\intercal}(i)U_{kk}^{\intercal}(i)U_{kk}(i)T_k(i), \tag{107}$$

which is a symmetric positive semi-definite matrix. Let $\{\widetilde{u}_{kk}^m(i)\}_{m=1}^{M_k}$ be entries on the main diagonal of $\widetilde{U}_{kk}(i)$. Then, it follows that $\widetilde{u}_{kk}^m(i) \geq 0$ for all $m = 1, \ldots, M_k$. From the L.H.S. of (61), we have

$$\begin{aligned}
&\operatorname{Tr}(U_{kk}(i)\left(X_{kk}(i) + \sigma_{n,k}^2(i)I_{M_k}\right)^{-1}U_{kk}^{\intercal}(i)) \\
&= \operatorname{Tr}(U_{kk}(i)T_k(i)\big(\Lambda_k(i) + \sigma_{n,k}^2(i)I_{M_k}\big)^{-1}T_k^{\intercal}(i)U_{kk}^{\intercal}(i)) \\
&\overset{\text{(a)}}{=} \operatorname{Tr}(\big(\Lambda_k(i) + \sigma_{n,k}^2(i)I_{M_k}\big)^{-1}\widetilde{U}_{kk}(i)) \\
&= \sum_{m=1}^{M_k} \frac{\widetilde{u}_{kk}^m(i)}{\lambda_k^m(i) + \sigma_{n,k}^2(i)} \\
&\leq \sum_{m=1}^{M_k} \frac{\widetilde{u}_{kk}^m(i)}{\sigma_{n,k}^2(i)} \\
&= \frac{\operatorname{Tr}(\widetilde{U}_{kk}(i))}{\sigma_{n,k}^2(i)} \\
&\overset{(107)}{=} \frac{\operatorname{Tr}(T_k^{\intercal}(i)U_{kk}^{\intercal}(i)U_{kk}(i)T_k(i))}{\sigma_{n,k}^2(i)} \\
&= \frac{\operatorname{Tr}(U_{kk}^{\intercal}(i)U_{kk}(i))}{\sigma_{n,k}^2(i)},
\end{aligned}$$

where in step (a) we used the property $\operatorname{Tr}(AB) = \operatorname{Tr}(BA)$ for any square matrices A and B of compatible sizes. The theorem now follows from (61).

REFERENCES

[1] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4129 – 4144, Aug. 2014.

[2] ——, "Diffusion LMS over multitask networks," *IEEE Trans. Signal Process.*, vol. 63, no. 11, pp. 2733 – 2748, Jun. 2015.

[3] M. Leng, W. P. Tay, T. Q. S. Quek, and H. Shin, "Distributed local linear parameter estimation using Gaussian SPAWN," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 244 – 257, Jan. 2015.

[4] J. Plata-Chaves, N. Bogdanović, and K. Berberidis, "Distributed diffusion-based LMS for node-specific adaptive parameter estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 3448 – 3460, Jul. 2015.

[5] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Proximal multitask learning over networks with sparsity-inducing coregularization," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6329 – 6344, Dec. 2016.

[6] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks with common latent representations," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 3, pp. 563 – 579, Apr. 2017.

[7] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Diffusion LMS for multitask problems with local linear equality constraints," *IEEE Trans. Signal Process.*, vol. 65, no. 19, pp. 4979 – 4993, Oct. 2017.

[8] Y. Wang, W. P. Tay, and W. Hu, "A multitask diffusion strategy with optimized inter-cluster cooperation," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 3, pp. 504 – 517, Apr. 2017.

[9] M. Chan, E. Campo, D. Estève, and J.-Y. Fourniols, "Smart homes – Current features and future perspectives," *Maturitas*, vol. 64, no. 2, pp. 90 – 97, Oct. 2009.

[10] A. Whitmore, A. Agarwal, and L. Da Xu, "The Internet of Things – A survey of topics and trends," *Information Systems Frontiers*, vol. 17, no. 2, pp. 261 – 274, Apr. 2015.

[11] S. H. Shah and I. Yaqoob, "A survey: Internet of Things (IoT) technologies, applications and challenges," in *Proc. IEEE Smart Energy Grid Engineering*, Oshawa, ON, Canada, Aug. 2016.

[12] S. Li, L. Da Xu, and S. Zhao, "5G Internet of Things: A survey," *J. of Ind. Inf. Integration*, vol. 10, pp. 1 – 9, Jun. 2018.

[13] W. Hu and W. P. Tay, "Multi-hop diffusion LMS for energy-constrained distributed estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 15, pp. 4022 – 4036, Aug. 2015.

[14] W. P. Tay, J. N. Tsitsiklis, and M. Z. Win, "Data fusion trees for detection: Does architecture matter?" *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 4155 – 4168, Sep. 2008.

[15] W. P. Tay, "Whose opinion to follow in multihypothesis social learning? A large deviations perspective," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 2, pp. 344 – 359, Mar. 2015.

[16] J. Ho, W. P. Tay, T. Q. Quek, and E. K. Chong, "Robust decentralized detection and social learning in tandem networks," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5019 – 5032, Oct. 2015.

[17] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *Proc. IEEE Symposium on Security and Privacy (SP)*, San Francisco, USA, Sep. 2019.

[18] L. Xie, I. M. Baytas, K. Lin, and J. Zhou, "Privacy-preserving distributed multi-task learning with asynchronous updates," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Halifax, Canada, Aug. 2017.

[19] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[20] V. Smith, C.-K. Chiang, M. Sanjabi, and A. Talwalkar, "Federated multi-task learning," in *Proc. 31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, USA, Dec. 2017.

[21] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," *arXiv preprint arXiv:2003.02133*, 2020.

[22] Y. Wang, X. Wu, and H. Donghui, "Using randomized response for differential privacy preserving data collection," in *Proc. ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Washington, D.C., USA, Aug. 2003.

[23] S. Xiong, A. D. Sarwate, and N. B. Mandayam, "Randomized requantization with local differential privacy," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Shanghai, China, Mar. 2016.

[24] A. D. Sarwate and L. Sankar, "A rate-disortion perspective on local differential privacy," in *Proc. Allerton Conf. on Commun., Control and Computing*, Monticello, IL, USA, Sep. 2014.

[25] J. Liao, L. Sankar, F. P. Calmon, and V. Y. Tan, "Hypothesis testing under maximal leakage privacy constraints," in *Proc. IEEE Int. Symp. on Inform. Theory*, Aachen, Germany, Jun. 2017.

[26] H. Imtiaz and A. D. Sarwate, "Differentially private distributed principal component analysis," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Calgary, Alberta, Canada, Apr. 2018.

[27] X. Ren, C.-M. Yu, W. Yu, S. Yang, X. Yang, J. A. McCann, and S. Y. Philip, "LoPub: High-dimensional crowdsourced data publication with local differential privacy," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 9, pp. 2151 – 2166, Mar. 2018.

[28] B. Razeghi and S. Voloshynovskiy, "Privacy-preserving outsourced media search using secure sparse ternary codes," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Calgary, Alberta, Canada, Apr. 2018.

[29] F. D. McSherry, "Privacy integrated queries: An extensible platform for privacy-preserving data analysis," in *Proc. ACM SIGMOD Inter. Conf. Management of data*, Providence, Rhode Island, USA, Jun. 2009.

[30] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Annual Inter. Conf. on the Theory and Applications of Cryptographic Techniques*, St. Petersburg, Russia, May 2006.

[31] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *Proc. IEEE Symp. on Foundations of Computer Science*, Monticello, IL, USA, Oct. 2013.

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

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

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

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

[36] X. Wang, J. He, P. Cheng, and J. Chen, "Privacy preserving collaborative computing: Heterogeneous privacy guarantee and efficient incentive mechanism," *IEEE Trans. Signal Process.*, vol. 67, no. 1, pp. 221 – 233, Jan. 2019.

[37] M. Sun, W. P. Tay, and X. He, "Toward information privacy for the Internet of Things: A non-parametric learning approach," *IEEE Trans. Signal Process.*, vol. 66, no. 7, pp. 1734 – 1747, Apr. 2018.

[38] X. He, W. P. Tay, H. Lei, M. Sun, and Y. Gong, "Privacy-aware sensor network via multilayer nonlinear processing," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10 834 – 10 845, Dec. 2019.

[39] M. Sun and W. P. Tay, "Decentralized detection with robust information privacy protection," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 85 – 99, 2020.

[40] ——, "On the relationship between inference and data privacy in decentralized IoT networks," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 852 – 866, 2020.

[41] C. X. Wang, Y. Song, and W. P. Tay, "Arbitrarily strong utility-privacy tradeoff in multi-agent systems," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 671 – 684, 2021.

[42] K. Diamantaras and S. Kung, "Data privacy protection by kernel subspace projection and generalized eigenvalue decomposition," in *IEEE Int. Workshop Machine Learning for Signal Processing*, Vietri sul Mare, Italy, Sep. 2016.

[43] S. Y. Kung, "Compressive privacy from information estimation," *IEEE Signal Process. Mag.*, vol. 34, no. 1, pp. 94 – 112, Jan. 2017.

[44] ——, "A compressive privacy approach to generalized information bottleneck and privacy funnel problems," *Journal of the Franklin Institute*, vol. 355, no. 4, pp. 1846 – 1872, Mar. 2018.

[45] Y. Song, C. X. Wang, and W. P. Tay, "Privacy-aware Kalman filtering," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Calgary, Canada, Apr. 2018.

[46] C. X. Wang, Y. Song, and W. P. Tay, "Preserving parameter privacy in sensor networks," in *Proc. IEEE Global Conf. on Signal and Information Processing*, Anaheim, USA, Nov. 2018.

[47] Y. Song, C. X. Wang, and W. P. Tay, "Compressive privacy for a linear dynamical system," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 895 – 910, 2020.

[48] T. S. Lau and W. P. Tay, "Privacy-aware quickest change detection," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Barcelona, Spain, May 2020.

[49] X. Zhao and A. H. Sayed, "Distributed clustering and learning over networks," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 3285 – 3300, Jul. 2015.

[50] J. Chen and A. H. Sayed, "Distributed Pareto optimization via diffusion strategies," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 205 – 220, Apr. 2013.

[51] A. H. Sayed, "Adaptation, learning, and optimization over networks," *Foundations and Trends in Machine Learning*, vol. 7, no. 4-5, pp. 311 – 801, 2014. [Online]. Available: http://dx.doi.org/10.1561/2200000051

[52] R. Nassif, C. Richard, J. Chen, A. Ferrari, and A. H. Sayed, "Diffusion LMS over multitask networks with noisy links," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Shanghai, China, Mar. 2016.

[53] I. E. K. Harrane, R. Flamary, and C. Richard, "Toward privacy-preserving diffusion strategies for adaptation and learning over networks," in *Proc. of European Signal Processing Conference (EUSIPCO)*, Budapest, Hungary, Aug. 2016.

[54] C. Wang, W. P. Tay, Y. Wang, and Y. Wei, "A privacy-preserving diffusion strategy over multitask networks," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Brighton, UK, May 2019.

[55] A. Khalili, M. A. Tinati, A. Rastegarnia, and J. A. Chambers, "Steady-state analysis of diffusion LMS adaptive networks with noisy links," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 974 – 979, Feb. 2012.

[56] X. Zhao, S.-Y. Tu, and A. H. Sayed, "Diffusion adaptation over networks under imperfect information exchange and non-stationary data," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3460 – 3475, Jul. 2012.

[57] X. Zhao and A. H. Sayed, "Clustering via diffusion adaptation over networks," in *Proc. Int. Workshop Cognitive Information Processing*, Baiona, Spain, May 2012.

[58] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 2016.

[59] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[60] A. H. Sayed, *Adaptive Filters*. New York, USA: Wiley-IEEE Press, 2008.

[61] G. T. Duncan, S. E. Fienberg, R. Krishnan, R. Padman, and S. F. Roehrig, "Disclosure limitation methods and information loss for tabular data," *Confidentiality, disclosure and data access: Theory and practical applications for statistical agencies*, pp. 135 – 166, 2001.

[62] J.-J. Salazar-González, "Statistical confidentiality: Optimization techniques to protect tables," *Computers & Operations Research*, vol. 35, no. 5, pp. 1638 – 1651, 2008.

[63] T. Evans, L. Zayatz, and J. Slanta, "Using noise for disclosure limitation of establishment tabular data," *Journal of Official Statistics*, vol. 14, no. 4, pp. 537 – 551, 1998.

Chengcheng Wang (S'17–M'18) received the B.Eng. degree in Electrical Engineering and Automation, and the Ph.D. degree in Control Science and Engineering from the College of Automation, Harbin Engineering University, Harbin, China, in 2011 and 2017, respectively. She is currently a Research Fellow in the School of Electrical and Electronic Engineering at Nanyang Technological University, Singapore. From September 2014 to September 2016, she was a visiting graduate researcher in the Adaptive Systems Laboratory at the University of California, Los Angeles, CA, USA. Her research interests include adaptive and statistical signal processing, distributed adaptation and learning, and federated learning.

Wee Peng Tay (S'06–M'08–SM'14) received the B.S. degree in Electrical Engineering and Mathematics, and the M.S. degree in Electrical Engineering from Stanford University, Stanford, CA, USA, in 2002. He received the Ph.D. degree in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2008. He is currently an Associate Professor in the School of Electrical and Electronic Engineering at Nanyang Technological University, Singapore. Dr. Tay received the Tan Chin Tuan Exchange Fellowship in 2015. He is a coauthor of the best student paper award at the Asilomar conference on Signals, Systems, and Computers in 2012, and the IEEE Signal Processing Society Young Author Best Paper Award in 2016. He was an Associate Editor for the IEEE Transactions on Signal Processing (2015 – 2019), and is currently an Associate Editor for the IEEE Transactions on Signal and Information Processing over Networks, an Editor for the IEEE Transactions on Wireless Communications, and an Editor for the IEEE Open Journal of Vehicular Technology. His research interests include information and signal processing over networks, distributed inference and estimation, privacy for IoT, machine learning, and applied probability.

Ye Wei (S'19–M'20) received the B.Eng. degree in Electrical Engineering and Automation, and the Ph.D. degree in Control Science and Engineering from the College of Automation, Harbin Engineering University, Harbin, China, in 2012 and 2019, respectively. He is currently an Assistant Professor in the School of Automation Engineering at Northeast Electric Power University, Jilin, China. From October 2017 to October 2018, he was a visiting student in the School of Computer Science and Engineering at Nanyang Technological University, Singapore. From October 2015 to October 2016, he was a visiting student in the Department of Electrical and Computer Engineering at Missouri University of Science and Technology, USA. His research interests include adaptive filtering, sparse signal recovery, and machine learning.

Yuan Wang received the B.Eng. degree from Northwestern Polytechnical University, Xi'an, China, in 2010, and M.Sc. and Ph.D. degrees from Nanyang Technological University, Singapore, in 2012 and 2019, respectively. He is currently a research scientist with the Department of Computing & Intelligence, Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore. His research interests include decentralized machine learning, federated learning, and deep learning.