INFERRING PRIVATE INFORMATION IN WIRELESS SENSOR NETWORKS

Daniel A. Burbano-L.[1], Jemin George[2], Randy A. Freeman[1], and Kevin M. Lynch[3]

[1]Department of Electrical Engineering and Computer Science
Northwestern University, Evanston, IL 60208-3118, USA
[2]U.S. Army Research Laboratory, Adelphi, MD 20783, USA
[3]Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208-3111

ABSTRACT

In wireless sensor networks, estimating a global parameter from locally obtained measurements via local interactions is known as the distributed parameter estimation problem. Solving these problems often require the deployment of distributed optimization algorithms that rely on a constant exchange of information among the sensor nodes. This makes such distributed algorithms vulnerable to attackers or malicious nodes that want to gain access to private information regarding the network. Based on the sliding mode control scheme, here we present a novel approach to infer sensitive information (e.g., gradient or private parameters of the local objective function) regarding a node of interest by intercepting the communication between the nodes. The effectiveness of the proposed approach is illustrated in a representative example of distributed event localization using an acoustic sensor network.

Index Terms— Distributed parameter estimation, wireless sensor network, distributed optimization, sliding control

1. INTRODUCTION

Wireless sensor networks have become a fundamental component in many practical applications like distributed environmental modeling [1], IoT (Internet of Things) [2], and localization problems [3]. Often but not always, the distributed strategies deployed over these networks are vulnerable to attackers that want to gain access to sensitive information [4]. A particular problem in wireless sensor networks is the *distributed parameter estimation*, where a group of sensors estimate a global parameter say $\boldsymbol{\theta} \in \mathbb{R}^n$ from local measurements by exchanging information with their nearest neighbors. The distributed parameter estimation has been used in applications like sensor localization [5] and event detection [6].

Consider a group of $N > 1$ agents equipped with sensors, each taking measurements $\boldsymbol{\phi}_i \in \mathbb{R}^n$ given by

$$\boldsymbol{\phi}_i = \boldsymbol{h}_i(\boldsymbol{\theta}, \boldsymbol{p}_i) + \boldsymbol{\xi}_i, \text{ for all } i \in \mathcal{N} := \{1, \cdots, N\} \tag{1}$$

where $\boldsymbol{h}_i : \mathbb{R}^{m=n+q} \mapsto \mathbb{R}^n$ is a continuously differentiable nonlinear function denoting the sensor mapping of the ith agent, which is assumed to be locally known, $\boldsymbol{\theta} \in \mathbb{R}^n$ denotes the unknown global variable, $\boldsymbol{p}_i \in \mathbb{R}^q$ is a vector of locally known (private) parameters, and $\boldsymbol{\xi}_i \in \mathbb{R}^n$ is a zero-mean, independent Gaussian noise with known variance $\boldsymbol{R}_i$, i.e., $\boldsymbol{\xi}_i \sim G(\mathbb{0}, \boldsymbol{R}_i)$

As an example, consider a network of acoustic sensors with bearing measurements ϕ_i to an event as illustrated in Figure 1. In this example, $\boldsymbol{\theta}$ is a two-dimensional vector representing the (X,Y) position of the event, and $\boldsymbol{p}_i$ is a two-dimensional vector whose entries are the location of the sensors, which is a sensitive information.

An estimate of $\boldsymbol{\theta}$ can be distributively obtained using the maximum likelihood method, which is equivalent to solving a distributed optimization problem [6] (see Section 2.3 for further details). Classical solutions to such optimization methods [7] require each agent to share the gradients of local objective functions that explicitly depend on $\boldsymbol{h}_i$ thus potentially leaking sensitive information like the private parameters $\boldsymbol{p}_i$. This is a common problem in distributed algorithms where a malicious agent wants access to private information regarding other agents in the network. To improve the privacy of sensitive data, a second-order PI (Proportional-Integral) optimization strategy has been proposed in [8] where agents only share local estimates rather than the gradients. Similar distributed optimization algorithms that only require the exchange of local estimates rather than gradients include the distributed subgradient methods [9, 10], distributed Alternating Direction Method of Multipliers [11, 12], and exact first-order algorithm (EXTRA) [13]. However, the complete privacy properties of these algorithms have not been fully explored.

This paper presents a novel methodology for inferring the private information (local gradients and parameters $\boldsymbol{p}_i$) of a target agent in the network by intercepting the incoming and outgoing communications. Inspired by sliding control [14], we devise an estimator to reconstruct the local gradients. Estimates of the gradient are then used to estimate the private vector $\boldsymbol{p}_i$ by solving a nonlinear least-squares problem. We show that the privacy property of the PI-optimization strategy proposed in [8] can be easily violated under minor assumptions thus allowing the malicious agents to obtain private information of the networked agents. In particular, we show that the proposed strategy is able to reconstruct the sensor location from the intercepted communications in the distributed event localization example shown in Figure 1.

This paper is organized as follows. The notation and preliminaries are given in Section 2, while the main result is presented

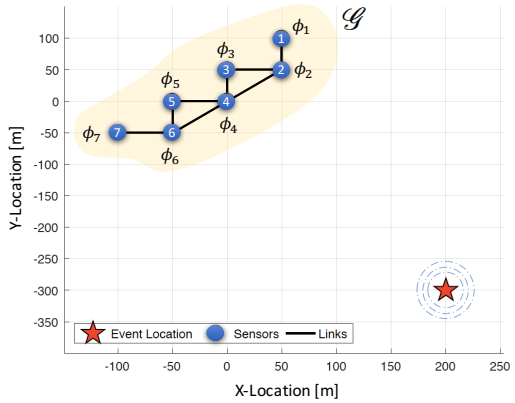

Fig. 1. Network of $N=7$ agents (blue circles) with acoustic sensors cooperating over a communication network (black links) to infer the shooter location (T_x,T_y) indicated by the red star. The network topology is represented by an undirected and connected graph $\mathscr{G}$.

in Section 3. In Section 4, we illustrate the effectiveness of our reconstruction strategy via numerical example, and concluding remarks appear in Section 5.

2. NOTATION AND PROBLEM STATEMENT

2.1. Notation

$|x|$ denotes the absolute value of any scalar $x\in\mathbb{R}$, while for any vector $\boldsymbol{x}\in\mathbb{R}^n$, $|\boldsymbol{x}|:=[|x_1|,\cdots,|x_N|]^\top$. The Euclidean norm for vectors is denoted by $\|\cdot\|_2$. $\mathbb{0}$ is a matrix of zeros of appropriate dimension. $\text{sgn}\{x\}$ for $x\in\mathbb{R}$ denotes the signum function, and for all $\boldsymbol{x}=[x_1,\cdots,x_n]^\top\in\mathbb{R}^n$, $\textbf{sgn}\{\boldsymbol{x}\}:=[\text{sgn}\{x_1\},\cdots,\text{sgn}\{x_n\}]^\top$. A *graph* $\mathscr{G}$ is defined by $\mathscr{G}=(\mathcal{V},\mathcal{E})$, where $\mathcal{V}=\{v_1,v_2,\cdots,v_N\}$ is the finite set of N nodes and $\mathcal{E}\subset\mathcal{N}\times\mathcal{N}$ is the set containing E edges between the nodes (v_i,v_j) for any $i,j\in\mathcal{N}$. An edge between node v_i and v_j is denoted by the ordered pair $e_{ij}=(v_i,v_j)$, which has an associated weight denoted by $a_{ij}\in\mathbb{R}$ for all $i,j\in\mathcal{N}$. We assume $\mathscr{G}$ to be *undirected and unweighted*; that is, if all edges are bidirectional with unit weight (i.e., if there exists an edge e_{ij}, then there is also an edge e_{ji}, and $a_{ij}=a_{ji}=1$). The *Adjacency matrix* $\boldsymbol{\mathcal{A}}=[a_{ij}]\in\mathbb{R}^{N\times N}$ of a graph $\mathscr{G}$ is defined by $\boldsymbol{\mathcal{A}}[a_{ij}]=1$ if there is an edge e_{ij} between nodes i and j, and $\boldsymbol{\mathcal{A}}[a_{ij}]=0$ otherwise.

2.2. Sliding control

Here we present a quick overview of the sliding control and the real-time approximation of the equivalent control. Consider the first order system

$$\dot{\sigma}(t)=a(t)+u_s(t),\;\; u_s(t)=-\kappa\text{sgn}\{\sigma(t)\} \tag{2}$$

where $\sigma(t)\in\mathbb{R}$ denotes the system state, $a(t)$ is an unknown time-varying but bounded disturbance, and $u_s(t)$ is the sliding control input. It is well known that selecting a sufficiently large κ ($\kappa>a_{\max}=\max_{t\geq0}|a(t)|$) guarantees that $\sigma(t)$ converges to zero in finite time [14]. Since $\sigma(t)$ reaches zero in finite time and remains there for all future time ($\sigma(t)\equiv0$), we have that the equivalent control input, $u_s\equiv u_{eq}$ satisfies $u_{eq}=-a(t)$ [14]. Thus, an estimate of $a(t)$ can be obtained by low-pass filtering the control input u_s [15], i.e., $\dot{\widehat{a}}=-(1/\tau)\widehat{a}+(1/\tau)u_s$, where τ is an arbitrary small positive constant.

2.3. Distributed Localization

An estimate $\widehat{\boldsymbol{\theta}}\in\mathbb{R}^n$ of the unknown target location $\boldsymbol{\theta}$ can be obtained using the Maximum likelihood method [6], which is equivalent to minimize the negative log-likelihood function $J(\boldsymbol{\theta})$, i.e.,

$$\min_{\boldsymbol{\theta}}\;\; J(\boldsymbol{\theta}),\;\; J(\boldsymbol{\theta}):=\sum_{i=1}^{N}\boldsymbol{f}_i(\boldsymbol{\theta},\boldsymbol{p}_i) \tag{3}$$

$$\boldsymbol{f}_i(\boldsymbol{\theta},\boldsymbol{p}_i):=\frac{1}{2}(\boldsymbol{\phi}_i-\boldsymbol{h}_i(\boldsymbol{\theta},\boldsymbol{p}_i))^\top\boldsymbol{R}_i^{-1}(\boldsymbol{\phi}_i-\boldsymbol{h}_i(\boldsymbol{\theta},\boldsymbol{p}_i)) \tag{4}$$

Note that each agent i has access to the local functions $\boldsymbol{f}_i(\boldsymbol{\theta},\boldsymbol{p}_i)$. Let $\widehat{\boldsymbol{\theta}}_i\in\mathbb{R}^n$ be the ith agent's local estimate of $\boldsymbol{\theta}$. Then, local solutions to the minimization problem (3) can be obtained in a distributed manner by using the Proportional-Integral (or second-order) gradient descent strategy [8]

$$\dot{\boldsymbol{v}}_i=\alpha\beta\sum_{j=1}^{N}a_{ij}(\widehat{\boldsymbol{\theta}}_i-\widehat{\boldsymbol{\theta}}_j), \tag{5}$$

$$\dot{\widehat{\boldsymbol{\theta}}}_i=-\alpha\boldsymbol{g}_i(\widehat{\boldsymbol{\theta}}_i,\boldsymbol{p}_i)-\boldsymbol{v}_i-\beta\sum_{j=1}^{N}a_{ij}(\widehat{\boldsymbol{\theta}}_i-\widehat{\boldsymbol{\theta}}_j), \tag{6}$$

with $\boldsymbol{v}_i(0)=\boldsymbol{v}_{i,o}\in\mathbb{R}^n$ and $\widehat{\boldsymbol{\theta}}_i(0)=\widehat{\boldsymbol{\theta}}_{i,o}\in\mathbb{R}^n$ being the the initial conditions of the ith agent, where $\boldsymbol{v}_{io}$ satisfies $\sum_{i=1}^{N}\boldsymbol{v}_{i,o}=\mathbb{0}$. In addition, α,β are positive constants and a_{ij} (for all $i,j\in\mathcal{N}$) are the elements of an *adjacency matrix* $\boldsymbol{\mathcal{A}}$ associated with the graph $\mathscr{G}$. Here $\mathscr{G}$ represents the communication topology between nodes. Moreover, the functions $\boldsymbol{g}_i(\widehat{\boldsymbol{\theta}}_i,\boldsymbol{p}_i)$ for all $i\in\mathcal{N}$ are the gradients of each local function $\boldsymbol{f}_i(\widehat{\boldsymbol{\theta}}_i,\boldsymbol{p}_i)$, which are given by

$$\boldsymbol{g}_i(\widehat{\boldsymbol{\theta}}_i,\boldsymbol{p}_i):=\left(\frac{\partial\boldsymbol{h}_i(\widehat{\boldsymbol{\theta}}_i,\boldsymbol{p}_i)}{\partial\widehat{\boldsymbol{\theta}}_i}\right)^\top\boldsymbol{R}_i^{-1}\left(\boldsymbol{h}_i(\widehat{\boldsymbol{\theta}}_i,\boldsymbol{p}_i)-\boldsymbol{\phi}_i\right) \tag{7}$$

Unlike the classic optimization approaches where the gradients $\boldsymbol{g}_i$ are communicated among agents [7], the strategy in (5)-(6) only requires communication of the local estimates $\widehat{\boldsymbol{\theta}}_j$, thus keeping the gradients and its parameters private.

2.4. Problem Formulation

Equations (5)-(6) can be rewritten as

$$\dot{\boldsymbol{v}}_i=\alpha\beta d_i\widehat{\boldsymbol{\theta}}_i-\alpha\beta\boldsymbol{u}_i \tag{8}$$

$$\dot{\widehat{\boldsymbol{\theta}}}_i=-\alpha\boldsymbol{g}_i(\widehat{\boldsymbol{\theta}}_i,\boldsymbol{p}_i)-\beta d_i\widehat{\boldsymbol{\theta}}_i-\boldsymbol{v}_i+\beta\boldsymbol{u}_i \tag{9}$$

where $d_i=\sum_{i=1}^{N}a_{ij}$ for all $i\in\mathcal{N}$ is the *node degree* for each agent and $\boldsymbol{u}_i:=\sum_{j=1}^{N}a_{ij}\widehat{\boldsymbol{\theta}}_j$ represent the incoming communication to the ith agent from its neighbours.

Problem 1. *For an agent of interest $k\in\mathcal{N}$, infer (or reconstruct) the kth gradient $\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k,\boldsymbol{p}_k)$ along with the private vector of parameters $\boldsymbol{p}_k$ by listening to (or intercepting) both the state $\widehat{\boldsymbol{\theta}}_k$ and the inter-agent communications $\boldsymbol{u}_k$.*

3. RECONSTRUCTION STRATEGY

We first assume that the constants α, β, and d_i are known. Indeed, α and β are globally known so a malicious agent can provide this information. Moreover, d_i is the number of intercepted signals of the agent for which we are interested in obtaining its private information.

We split the reconstruction problem in two; namely, (i) gradient reconstruction and (ii) parameter reconstruction.

3.1. Gradient Reconstruction

Based on the fact that the filtered version of the equivalent control in the sliding mode strategy can provide an estimate of an unknown term (see Section 2.2), here we propose the following estimator for reconstructing the gradient $\boldsymbol{g}_k$ of a node of interest

$$\dot{\widehat{\boldsymbol{v}}}=\alpha\beta d_i\widehat{\boldsymbol{\theta}}_k-\alpha\beta\boldsymbol{u}_k,\ \widehat{\boldsymbol{v}}(0)=\mathbb{0}, \tag{10}$$

$$\dot{\boldsymbol{z}}=-\widehat{\boldsymbol{v}}-\widehat{\boldsymbol{a}}-\beta d_i\widehat{\boldsymbol{\theta}}_k+\beta\boldsymbol{u}_i,\ \boldsymbol{z}(0)=\widehat{\boldsymbol{\theta}}_{k,o}, \tag{11}$$

$$\dot{\widehat{\boldsymbol{a}}}=-\frac{1}{\tau}\widehat{\boldsymbol{a}}-\frac{\kappa}{\tau}\mathbf{sgn}\{\widehat{\boldsymbol{\theta}}_k-\boldsymbol{z}\},\ \widehat{\boldsymbol{a}}(0)=\widehat{\boldsymbol{a}}_o \tag{12}$$

where $\widehat{\boldsymbol{v}}\in\mathbb{R}^n$ and $\widehat{\boldsymbol{z}}\in\mathbb{R}^n$ are the estimates of $\boldsymbol{v}_k$ and $\widehat{\boldsymbol{\theta}}_k$ respectively, and $\widehat{\boldsymbol{a}}\in\mathbb{R}^n$ is an estimate of the unknown gradient $\boldsymbol{g}_k$ plus a constant bias. τ is an arbitrary small positive constant, and $\kappa>0$ is a feedback gain that is typically large.

Theorem 1. *Assume the distributed optimization algorithm* (5)-(6) *has bounded solutions and the local gradients $\boldsymbol{g}_i$ of each agent are continuous. In addition, assume that the inter-agent communications $\boldsymbol{u}_k$ and $\widehat{\boldsymbol{\theta}}_k$ of the kth agent are intercepted. Then, by choosing an arbitrary large value of κ, $\widehat{\boldsymbol{a}}$ in equation* (12) *with $0<\tau\ll 1$ converges to the private gradient $\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}},\boldsymbol{p}_i)+\boldsymbol{v}_{k,o}$ in finite time $t^*>0$; that is, for all $t\geq t^*$*

$$\|\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k(t),\boldsymbol{p}_k)+\boldsymbol{v}_{k,o}-\widehat{\boldsymbol{a}}(t)\|_2=\mathcal{O}(\tau) \tag{13}$$

where $\mathcal{O}(\tau)$ is a residual error.

Proof. We start by calculating the error dynamics by setting $\boldsymbol{\sigma}=\widehat{\boldsymbol{\theta}}_k-\boldsymbol{z}$, thus yielding

$$\dot{\boldsymbol{\sigma}}=-\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k,\boldsymbol{p}_k)-(\boldsymbol{v}-\widehat{\boldsymbol{v}})+\widehat{\boldsymbol{a}}. \tag{14}$$

Then, integrating both sides of equations (8) and (10) from 0 to t, we find that $\boldsymbol{v}(t)$ and $\widehat{\boldsymbol{v}}(t)$ satisfies

$$\boldsymbol{v}(t)=\int_0^t(\alpha\beta d_k\widehat{\boldsymbol{\theta}}_k(s)-\alpha\beta\boldsymbol{u}_k(s))ds+\boldsymbol{v}_{k,o} \tag{15}$$

$$\widehat{\boldsymbol{v}}(t)=\int_0^t(\alpha\beta d_k\widehat{\boldsymbol{\theta}}_k(s)-\alpha\beta\boldsymbol{u}_k(s))ds \tag{16}$$

Thus, $\boldsymbol{v}(t)-\widehat{\boldsymbol{v}}(t)=\boldsymbol{v}_{k,o}$ for all $t\geq 0$ and

$$\dot{\boldsymbol{\sigma}}=-\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k,\boldsymbol{p}_k)-\boldsymbol{v}_{k,o}+\widehat{\boldsymbol{a}} \tag{17}$$

From (12) we have that $\tau\dot{\widehat{\boldsymbol{a}}}=-\widehat{\boldsymbol{a}}-\kappa\mathbf{sgn}\{\boldsymbol{\sigma}\}$. Because τ is assumed to be arbitrarily small, according to perturbation theory [16] $\tau\dot{\widehat{\boldsymbol{a}}}\approx 0$ and we have that $\widehat{\boldsymbol{a}}=-\kappa\mathbf{sgn}\{\boldsymbol{\sigma}\}$. Substituting $\widehat{\boldsymbol{a}}$ in equation (17) yields

$$\dot{\boldsymbol{\sigma}}=\boldsymbol{a}(t)+\boldsymbol{u}_s, \tag{18}$$

$$\boldsymbol{a}(t):=-(\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k,\boldsymbol{p}_k)+\boldsymbol{v}_{k,o}),\ \boldsymbol{u}_s:=-\kappa\mathbf{sgn}\{\boldsymbol{\sigma}\}. \tag{19}$$

Next, considering the Lyapunov candidate function $V=(1/2)\boldsymbol{\sigma}^\top\boldsymbol{\sigma}$, yields $\dot{V}=\boldsymbol{\sigma}^\top\boldsymbol{a}(t)-\kappa|\boldsymbol{\sigma}|$. Because, the solution of the optimization problem in (5)-(6) is bounded, from the continuity of $\boldsymbol{g}_k$ we have that $\|\boldsymbol{a}(t)\|\leq c<\infty$ for all $t>0$. Then, $\dot{V}\leq c|\boldsymbol{\sigma}|-\kappa|\boldsymbol{\sigma}|$. Choosing a large value of κ guarantees $\dot{V}\leq 0$ and $\boldsymbol{\sigma}$ converges to zero in finite time [14]. Once $\boldsymbol{\sigma}(t)$ reaches zero it remains there (slides) for all future times ($\boldsymbol{\sigma}(t)\equiv 0$). According to the equivalent control method, we have that control input, $\boldsymbol{u}_s\equiv\boldsymbol{u}_{eq}$ satisfies $\boldsymbol{u}_{eq}=-\boldsymbol{a}(t)$ once sliding is attained [14]. Thus, $\widehat{\boldsymbol{a}}=\boldsymbol{u}_{eq}=-\boldsymbol{a}(t)$ and the proof is complete. □

Note that the residual error $\mathcal{O}(\tau)$ can be made arbitrarily small by decreasing the value of τ.

3.2. Reconstruction of Private Parameters

From Theorem 1 we have that $\widehat{\boldsymbol{a}}$ is an estimate of the unknown term $\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k,\boldsymbol{p}_k)+\boldsymbol{v}_{k,o}$ and the estimation error converges in finite time $t^*>0$. That is, for all $t\geq t^*$, we have

$$\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k(t),\boldsymbol{p}_k)+\boldsymbol{v}_{k,o}\approx\widehat{\boldsymbol{a}}(t). \tag{20}$$

Let $[t_1,t_2]$ with $t_2>t_1\geq t^*$ be the interval of time where M measurements are taken of the signals $\widehat{\boldsymbol{a}}(sT)$ and $\widehat{\boldsymbol{\theta}}_k(sT)$ with sampling rate T and $s=\{1,\cdots,M\}$. Then, assuming the functional form of the gradient and the variance of noise $\boldsymbol{R}_k$ are known, the unknown parameters $\boldsymbol{v}_{k,o}$ and $\boldsymbol{p}_k$ along with the measurements $\boldsymbol{\phi}_k$ can be estimated by solving

$$\min_{\boldsymbol{v}_{k,o},\boldsymbol{p}_k,\boldsymbol{\phi}_k}\sum_{s=1}^{M}\|\alpha\boldsymbol{g}_k(\widehat{\boldsymbol{\theta}}_k(sT),\boldsymbol{p}_k)+\boldsymbol{v}_{k,o}-\widehat{\boldsymbol{a}}(sT)\|_2^2 \tag{21}$$

Then an standard optimization strategy like the optimization toolbox of Matlab can be used to solve (21). Is important to mention that an initial guess of the constant bias $\boldsymbol{v}_{k,o}$ can be made by calculating the Fourier transform of the signal $\widehat{\boldsymbol{a}}(t)$ for the interval $[t_1,t_2]$ and taking the amplitude of the Dirac delta function at zero frequency. Moreover, in some cases, estimating $\boldsymbol{\phi}_k$ can be avoided by directly manipulating the set of equations in (20) as illustrated in the application example in Section 4 (see equation (24)).

4. NUMERICAL RESULTS

Consider the problem of cooperative event localization depicted in Figure 1, where there is a network of $N=7$ ($\mathcal{N}=\{1,\cdots,7\}$) agents represented by an undirected and connected graph $\mathcal{G}$. Each

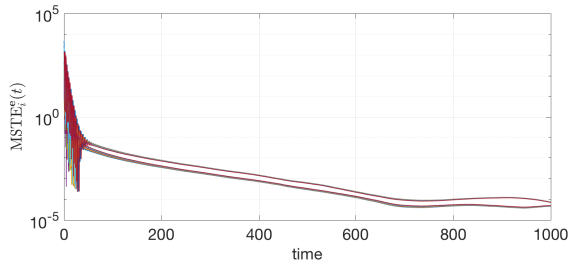

Fig. 2. Time evolution of $\mathrm{MSTE}_i^e(t)$ over 10^3 Monte Carlo simulations for all $i \in \mathcal{N}$.

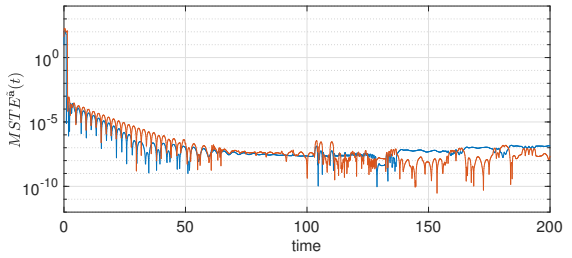

Fig. 3. Time evolution of the mean-square tracking error of $\widetilde{\boldsymbol{a}}(t) = \widehat{\boldsymbol{a}}(t) - \boldsymbol{a}(t)$ over 10^3 Monte Carlo simulations.

agent is equipped with an acoustic sensor consisting of an array of microphones that can obtain the direction of arrival of the acoustic signal (see [17, 18, 6] for further details). The goal is to provide an estimate of the shooter (or target) location represented by $\boldsymbol{\theta} = [T_x, T_y]^\top \in \mathbb{R}^2$. Each sensor takes a measurement $\phi_i \in \mathbb{R}$ according to (1), where the nonlinear function $h(\boldsymbol{\theta}, \boldsymbol{p}_i)$ is given by [6]

$$h(\boldsymbol{\theta}, \boldsymbol{p}_i) = \arctan\left(\frac{T_y - S_i^y}{T_x - S_i^x}\right) \tag{22}$$

with $\boldsymbol{p}_i = [S_i^x, S_i^y]^\top \in \mathbb{R}^2$ denoting the sensor locations, for all $i \in \mathcal{N}$. We consider the target location is given by $\boldsymbol{\theta} = [200, -300]^T$ (indicated by the red star in Figure 1), and the variance of measurement noise $R_i = R = 10^{-3}$, for all $i \in \mathcal{N}$. For the implementation of the distributed optimization algorithm in (5)-(6), we select $\alpha = 10$, $\beta = 0.5$, the initial conditions are chosen randomly, and the gradient $\boldsymbol{g}_i(\widehat{\boldsymbol{\theta}}, \boldsymbol{p}_i)$ is given by

$$\boldsymbol{g}_i(\widehat{\boldsymbol{\theta}}, \boldsymbol{p}_i) = \begin{bmatrix} -(T_y - S_i^y)/\Delta \\ (T_x - S_i^x)/\Delta \end{bmatrix} R^{-1}(\phi_i - h_i(\widehat{\boldsymbol{\theta}}_i, \boldsymbol{p}_i)), \tag{23}$$

where $\Delta = (S_i^x - T_x)^2 + (S_i^y - T_y)^2$. For the sake of completeness we first show that the distributed solution to the problem (3) is equivalent to the centralized one. To do so we solve (3) using the gradient flow $\dot{\widehat{\boldsymbol{\theta}}} = -\alpha \sum_{i=1}^N \boldsymbol{g}_i(\widehat{\boldsymbol{\theta}}, \boldsymbol{p}_i)$. We conducted 10^3 Monte Carlo simulations and we calculate the mean-square tracking error, $\mathrm{MSTE}_i^e(t) = (1/10^3) \sum_{l=1}^{10^3} \|\boldsymbol{e}_i\|^2$, with $\boldsymbol{e}_i = \hat{\boldsymbol{\theta}}(t) - \hat{\boldsymbol{\theta}}_i(t)$ denoting the difference between the centralized and distributed solutions. The time evolution of $\mathrm{MSTE}_i^e(t)$ is shown in Figure 2. Note that the distributed algorithm recovers the centralized solution.

Moreover, to illustrate the effectiveness of our reconstruction

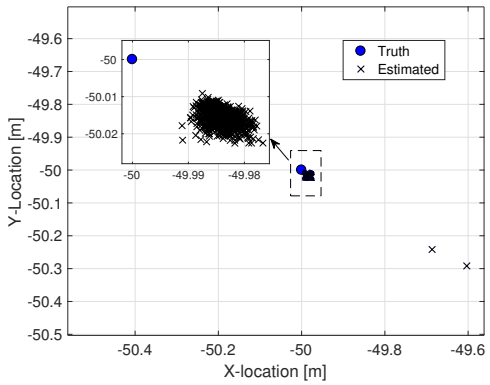

Fig. 4. Estimated sensor locations over 10^3 simulations.

strategy, we select node 6 as the node for which we are interested in recovering its private data. Note that the location of the sixth agent is given by $\boldsymbol{p}_i = [-50, -50]^\top$, and it is communicating to sensors 4, 5, and 7. Then, by intercepting $\widehat{\boldsymbol{\theta}}_{k=6}(t)$ and $\boldsymbol{u}_6$ we estimate the unknown gradient of the 6th node using the estimator (10)-(12) setting $\tau = 10^{-3}$ and $\kappa = 10^2$. Figure 3 shows the mean-square estimation error for the gradient estimates across the Monte Carlo simulations. Next, gradient estimates $\widehat{\boldsymbol{a}}$ from the time interval $[t_1 = 5, t_2 = 20]$ are used to estimate the sensor location and initial conditions $\boldsymbol{v}_{6,o}$. From (20), we have $\boldsymbol{a}(t) = [\widehat{a}_1(t), \widehat{a}_2(t)]^\top \approx \alpha \boldsymbol{g}_6(\widehat{\boldsymbol{\theta}}_6(t), \boldsymbol{p}_6) + \boldsymbol{v}_{6,o}$, where $\boldsymbol{g}_6$ is defined in (23). Then, by setting $\boldsymbol{v}_{6,o} = [v_{6,1o}, v_{6,2o}]^\top$ and $y(t) = (\widehat{a}_2(t) - v_{6,2o})/(\widehat{a}_1(t) - v_{6,1o})$, we find that $y(t) = -(T_x(t) - S_6^x)/(T_y(t) - S_6^y)$. Then, similar to the minimization problem in (21), the sensor locations of the sixth node $\boldsymbol{p}_i = [S_6^x, S_6^y]^\top$ can be find by solving

$$\min_{\boldsymbol{v}_{k,o}, \boldsymbol{p}_k} \sum_{s=1}^{M} \left| y(sT) + \frac{(T_x(sT) - S_6^x)}{(T_y(sT) - S_6^y)} \right|^2 \tag{24}$$

with $T = 10^{-1}$. Note that by manipulating the set of equations in (20) (or using the transformation $y(t)$), the estimation of ϕ_6 is avoided. The estimated sensor locations obtained from 10^3 Monte Carlo runs solving (24) are shown in Fig. 4. Note that the location is inferred with high accuracy. Indeed, the root-mean-square error calculated for both estimate of the sensor location and $\boldsymbol{v}_6$ are 2.5 and 0.0959 units, respectively.

5. CONCLUSION

We have proposed a novel methodology to infer private information in sensor networks by intercepting the communications of a distributed optimization strategy that is deployed over the network for solving a target localization problem. In particular, the local gradient of an agent of interest is estimated using sliding mode control. Then, the private parameters are inferred by solving a nonlinear least-squares problem. We showed the effectiveness of the proposed strategy via numerical simulations on an acoustic network. Our results can potentially provide important insights when designing more sophisticated and secure strategies [19, 20].

6. REFERENCES

[1] K. M. Lynch, I. B. Schwartz, P. Yang, and R. A. Freeman, "Decentralized environmental modeling by mobile sensor networks," *IEEE Transactions on Robotics*, vol. 24, no. 3, pp. 710–724, 2008.

[2] S. Li, L. Da Xu, and X. Wang, "Compressed sensing signal and data acquisition in wireless sensor networks and internet of things," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2177–2186, 2013.

[3] T. J. Chowdhury, C. Elkin, V. Devabhaktuni, D. B. Rawat, and J. Oluoch, "Advances on localization techniques for wireless sensor networks: A survey," *Computer Networks*, vol. 110, pp. 284–305, 2016.

[4] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2483–2493, 2013.

[5] G. Destino and G. Abreu, "On the maximum likelihood approach for source and network localization," *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4954–4970, 2011.

[6] J. George, "Distributed maximum likelihood using dynamic average consensus," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 3834–3838.

[7] M. Zinkevich, J. Langford, and A. J. Smola, "Slow learners are fast," in *Advances in neural information processing systems*, 2009, pp. 2331–2339.

[8] S. S. Kia, J. Cortés, and S. Martínez, "Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication," *Automatica*, vol. 55, pp. 254–264, 2015.

[9] A. Nedić and A. Ozdaglar, "Approximate primal solutions and rate analysis for dual subgradient methods," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1757–1780, 2009.

[10] A. Nedić, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 922–938, 2010.

[11] E. Wei and A. Ozdaglar, "Distributed alternating direction method of multipliers," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 5445–5450.

[12] W. Deng and W. Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," *Journal of Scientific Computing*, vol. 66, no. 3, pp. 889–916, Mar 2016.

[13] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXTRA: An Exact First-Order Algorithm for Decentralized Consensus Optimization," *ArXiv e-prints*, Apr. 2014.

[14] V. I. Utkin, *Sliding modes in control and optimization*. Springer Science & Business Media, 2013.

[15] C. Edwards and Y. Shtessel, "Dual-layer adaptive sliding mode control," in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 4524–4529.

[16] H. Khalil, *Nonlinear Systems*. Prentice Hall, 2002.

[17] J. George and L. M. Kaplan, "Shooter localization using soldier-worn gunfire detection systems," in *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, 2011, pp. 1–8.

[18] ——, "A finite point process approach to multi-target localization using transient measurements," *Information Fusion*, vol. 32, pp. 62–74, 2016.

[19] P. Braca, R. Lazzeretti, S. Marano, and V. Matta, "Learning with privacy in consensus + obfuscation," *IEEE Signal Processing Letters*, vol. 23, no. 9, pp. 1174–1178, 2016.

[20] I. E. K. Harrane, R. Flamary, and C. Richard, "Toward privacy-preserving diffusion strategies for adaptation and learning over networks," in *Signal Processing Conference (EUSIPCO), 2016 24th European*, 2016, pp. 1513–1517.