DATA INJECTION ATTACK ON DECENTRALIZED OPTIMIZATION

Sissi Xiaoxiao Wu[†], *Hoi-To Wai*[‡], *Anna Scaglione*[‡], *Angelia Nedić*[‡], *Amir Leshem*[⋆]

[†]Shenzhen University, College of Information Engineering, Shenzhen, China
[‡]School of ECEE, Arizona State Univ., Tempe, AZ, USA [⋆]Faculty of Engg., Bar-Ilan Univ., Israel.

ABSTRACT

This paper studies the security aspect of gossip-based decentralized optimization algorithms for multi agent systems against data injection attacks. Our contributions are two-fold. First, we show that the popular distributed projected gradient method (by *Nedić et al.*) can be attacked by *coordinated insider* attacks, in which the attackers are able to steer the final state to a point of their choosing. Second, we propose a metric that can be computed locally by the trustworthy agents processing their own iterates and those of their neighboring agents. This metric can be used by the trustworthy agents to detect and localize the attackers. We conclude the paper by supporting our findings with numerical experiments.

***Index Terms*—** Decentralized optimization, gossip algorithms, data injection attack.

1. INTRODUCTION

Decentralized multi-agent optimization has made significant strides in the past ten years. There is a large literature on multi-agent optimization algorithms, often referred as gossip-based or network diffusion algorithms, that rely on local computations and near neighbors communications to solve iteratively a wide class of constrained optimization problems, e.g., [1–14]. A key advantage of these parallel computation algorithms is the built-in fault tolerance to intermittent computation or communication due to normal failures, as the agents involved can reorganize themselves automatically sailing through these failures. There has been steady progress in expanding the class of problems amenable to a decentralized solution, and a lot of efforts have been made in improving their convergence rate and communication requirements. However, the issue of securing these algorithms against malicious data injection attacks has not received much attention until very recently [15–19]. Naturally, to prevent interference from unauthorized nodes, one can resort to authentication and encryption (see e.g. [20, 21]). However, in the case of an *insider attack*, gossip-based algorithms are highly vulnerable, even if only one node is compromised. In fact, it is relatively easy to show that gossip-based algorithms are vulnerable to data injection attacks and that such attacks, if coordinated, can steer the network to a final result of the attackers choosing [18, 22]. The flat, self-organizing architecture, which is the selling feature for these algorithms, during an attack becomes a liability.

This paper studies the effect of insider attacks on decentralized optimization algorithms. In particular, we focus on a classical distributed projected gradient (DPG) method introduced by *Nedić* et al. [5]. To this end, we first propose a new, stronger attack model that cannot be detected using the state-of-the-art protection method. This attack model is shown analytically to be always successful even when the underlying communication graph is time varying. We then propose a new metric that is locally computable by the trustworthy agents to detect and localize the attackers. The latter is shown analytically and empirically to perform the detection and localization tasks successfully. Notice that in [18], the authors proposed a robust consensus based distributed optimization algorithm that is guaranteed to converge to the convex hull of the union of the sets of minimizers of the normal nodes' objectives, which may not include a global optimum in general. On the other hand, with a similar algorithm, [19] proves that an optimal solution can be found under the assumption that the local functions admit a common optimal solution. Both papers did not discuss the task of detection and localization of inside attackers.

This work is supported by the National Natural Science Foundation of China under Grant 009989, the US National Science Foundation EAGER CCF 1553746, NSF CCF-BSF 1714672, and BSF Grant 2016660.

2. DISTRIBUTED PROJECTED GRADIENT

We study consensus-based optimization algorithms for tackling the following problem on an n-agents network:

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^d} f(\boldsymbol{\theta}) := (1/n)\sum_{i=1}^n f_i(\boldsymbol{\theta}) \text{ s.t. } \boldsymbol{\theta}\in\mathcal{C}\,, \tag{1}$$

where $\mathcal{C}\subseteq\mathbb{R}^d$ is a closed convex set and $f_i:\mathbb{R}^d\to\mathbb{R}$ is a differentiable function over $\mathcal{C}$. Here, f_i is a *private* function such that it is only known to the ith agent, e.g., it may correspond to the measurements made by the sensors of the ith agent. We let $f^\star$ be the optimal value of problem (1). At time $t\in\mathbb{N}$, the n agents are connected via an undirected time varying graph $G(t)=(V,E(t))$ where $V=[n]:=\{1,...,n\}$ and $E(t)\subseteq[n]\times[n]$ is the edge set. The graph is associated with a weighted adjacency matrix $\boldsymbol{W}(t)\in\mathbb{R}^{n\times n}$ where $[\boldsymbol{W}(t)]_{ij}:=W_{ij}(t)=0$ if $(j,i)\notin E(t)$. The union of these graphs is defined as $G:=(V,E)$ with $E:=\cup_{t=1}^\infty E(t)$. We assume the following on the sequence of graphs $\{G(t)\}_{t\geq1}$ and adjacency matrices $\{\boldsymbol{W}(t)\}_{t\geq1}$:

H1. *If $(i,j)\in E(t)$, then $W_{ij}(t)\geq\eta$ for some $\eta\in(0,1)$;*

H2. *There exists $B_0<\infty$ such that the graph $(V,\cup_{\ell=1}^{B_0}E(t+\ell))$ is connected.*

H3. *For all $t\geq1$, it holds that (a) $\boldsymbol{W}(t)\geq\mathbf{0}$, $\boldsymbol{W}(t)\mathbf{1}=\mathbf{1}$ and (b) $\boldsymbol{W}^\top(t)\mathbf{1}=\mathbf{1}$.*

The distributed projected gradient method (DPG) method [5] tackles (1) by performing the recursion:

$$\begin{aligned}\boldsymbol{\theta}_i(t+1)&=\mathcal{P}_{\mathcal{C}}\big(\bar{\boldsymbol{\theta}}_i(t)-\gamma(t)\nabla f_i\big(\bar{\boldsymbol{\theta}}_i(t)\big)\big)\,,\\ \bar{\boldsymbol{\theta}}_i(t)&=\textstyle\sum_{j=1}^n W_{ij}(t)\boldsymbol{\theta}_j(t)\,,\end{aligned} \tag{2}$$

for all $i\in[n]$ and $t\geq1$, where $\gamma(t)>0$ is a diminishing step size. For convex problems, it was shown in [5] the DPG method converges to an optimal solution of (1):

Fact 1. *Under H1, H2, H3(a) and H3(b). Suppose that each of f_i is convex, $\|\nabla f_i(\boldsymbol{\theta})\| \leq C$ for some C and for all $\boldsymbol{\theta} \in \mathcal{C}$, and the step size satisfies $\sum_{t=1}^{\infty} \gamma(t) = \infty$, $\sum_{t=1}^{\infty} \gamma^2(t) < \infty$, then for all $i, j \in [n]$ we have*

$$\lim_{t\to\infty} f(\boldsymbol{\theta}_i(t)) = f^\star \text{ and } \lim_{t\to\infty} \|\boldsymbol{\theta}_i(t) - \boldsymbol{\theta}_j(t)\| = 0 \,. \quad (3)$$

For non-convex problems, the same DPG method (2) is shown in [14] to converge to a KKT point of (1) under different assumptions on the sequence of adjacency matrices $\{\boldsymbol{W}(t)\}_{t=1}^{\infty}$.

3. COORDINATED ATTACK ON DPG

We next describe an attack scheme on the DPG method and show that a straightforward attack scheme can be successful under mild assumptions on the network topology. To set up the stage, let us define $A \subseteq V$, $A \neq \emptyset$ as the set of attackers and $N := V \setminus A$ as the set of trustworthy agents.

The attackers' goal is to steer the final state $\lim_{t\to\infty} \boldsymbol{\theta}_i(t)$ to a target state $\boldsymbol{x} \in \mathcal{C}$ for all agents in V, instead of converging to a stationary point of (1). To do so, the attackers follow a different update rule than (2), *i.e.,* we have

$$\boldsymbol{\theta}_j(t) = \boldsymbol{x} + \boldsymbol{z}_j(t), \ \forall\, j \in A \,; \quad (4)$$

meanwhile the trustworthy agents, *i.e.,* agent i with $i \in N$, apply the same DPG rule in Eq. (2). In the above, $\boldsymbol{z}_j(t)$ is an artificial noise introduced to obfuscate the trustworthy agents into believing that these agents are trustworthy as they appear to be converging. The artificial noise satisfies

H4. *For all $j \in A$, the artificial noise $\boldsymbol{z}_j(t)$ vanishes almost surely as $t \to \infty$,* i.e., $\lim_{t\to\infty} \|\boldsymbol{z}_j(t)\| =^{a.s.} 0$.

Notice that the attackers' states are not affected by the trustworthy agents' states at any time. Equivalently we can model the time varying graph such that for any $i \in N$ and $j \in A$, we have $(j, i) \notin E(t)$, *i.e.,* there is no information flow from any trustworthy agent to an attacker. For the attack to be successful, we require the following assumption which is mild. Let $E(N; t)$ be the edge set of the subgraph of $G(t)$ with only the nodes in N.

H 5. *There exists $B_1, B_2 < \infty$ such that for all $t \geq 1$, (a) the composite sub-graph $(N, \cup_{\ell=t+1}^{t+B_1} E(N; \ell))$ is connected; (b) there exists a pair $i \in N$, $j \in A$ with $(i, j) \in E(t) \cup \ldots \cup E(t + B_2 - 1)$.*

We show that:

Proposition 1. *Under H1, H3(a), H4 and H5. If the gradient is bounded such that $\|\nabla f_i(\boldsymbol{\theta})\| \leq M$ for some M and for all $\boldsymbol{\theta} \in \mathcal{C}$, and $\gamma(t) \to 0$, then:*

$$\lim_{t\to\infty} \max_{i\in N} \|\boldsymbol{\theta}_i(t) - \boldsymbol{x}\| =^{a.s.} 0 \,. \quad (5)$$

The proof is relegated to Appendix 5.1 and it involves analyzing the attacked DPG dynamics using a new result on the product of transition matrices of the time varying graphs involved. Notice that our result can be easily specialized to the case with a static transition matrix. Proposition 1 shows that the attackers can always succeed in steering the trustworthy agents to the attackers' desired vector.

Note that the attack strategy in (4) could be driven by the attacker trying to solve another instance of (1) with the DPG method, where the sum of the objective functions only extends to the attacker set A. If the optimum point of this instance of (1) is unique and given by $\boldsymbol{x}$, assuming that the sub-graph $G[A]$ satisfies H1 to H3, the deviation of the attackers' DPG iterates, $\boldsymbol{\theta}_j(t)$, from $\boldsymbol{x}$ forms a sequence $\boldsymbol{z}_j(t)$ that satisfies H4.

3.1. Detecting and Locating the Attackers

Our next endeavor is to detect the presence of attackers and to localize them. If successful, we can eliminate the influences from the attackers such that the attackers no longer steer the final state away from the optimal solution to (1). In the following, the proposed scheme requires accruing historical information about the iterates from K different instances of the DPG (2).

Naturally, we shall focus on the case when H1–H5 hold from the previous section such that the attack is successful. Concretely, at the kth instance, we apply the DPG to:

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^d} \ f^k(\boldsymbol{\theta}) = (1/n) \textstyle\sum_{i=1}^n f_i^k(\boldsymbol{\theta}) \ \text{ s.t. } \ \boldsymbol{\theta} \in \mathcal{C} \,, \quad (6)$$

and the iterates of DPG dynamics for the above are denoted using $\{\boldsymbol{\theta}_i^k(t)\}_{i=1}^n$. After running the DPG for K different instances, we consider the following difference vector:

$$\boldsymbol{\eta}_m := (1/K) \textstyle\sum_{k=1}^K \Big(\boldsymbol{\theta}_m^k(\infty) - \boldsymbol{\theta}_m^k(0)\Big) \,, \quad (7)$$

which is the difference between the initial value and the steady state value held by node, averaged over all K instances. Note that in practice, it is impossible to compute this metric at $t = \infty$, so we just choose a sufficiently large t at which the system approaches the steady state. It is important to note that for agent i, the vector above can be computed locally as long as $m \in \mathcal{N}_i^{\text{in}}$, *i.e.,* m is in the *in-neighbor* set of agent i where $\mathcal{N}_i^{\text{in}} := \{j : (j, i) \in E\}$, note that $E = \cup_{t=1}^{\infty} E(t)$ is the union of the edge set over an infinite horizon.

Detection Task. We propose to use the vector $\boldsymbol{\eta}_m$ for the *detection task*, which is a hypothesis test for:

$\mathcal{H}_0^i$ —there is no attacker in $\mathcal{N}_i^{\text{in}}$, *i.e.,* $A \cap \mathcal{N}_i^{\text{in}} = \emptyset$;

$\mathcal{H}_1^i$ —there exists an attacker in $\mathcal{N}_i^{\text{in}}$, *i.e.,* $A \cap \mathcal{N}_i^{\text{in}} \neq \emptyset$.

The detection task corresponds to —

$$\mathcal{D}^i := \frac{1}{|\mathcal{N}_i^{\text{in}}|} \sum_{m\in\mathcal{N}_i^{\text{in}}} \left| \mathbf{1}^\top \left(\boldsymbol{\eta}_m - \frac{1}{|\mathcal{N}_i^{\text{in}}|} \sum_{j\in\mathcal{N}_i^{\text{in}}} \boldsymbol{\eta}_j \right) \right| \underset{\mathcal{H}_1^i}{\overset{\mathcal{H}_0^i}{\lessgtr}} \delta_I \,, \quad (8)$$

where $\delta_I > 0$ is a predefined threshold. To explain the intuition behind the test (8), we observe that under both $\mathcal{H}_0^i$ and $\mathcal{H}_1^i$,

$$\boldsymbol{\eta}_m - \frac{1}{|\mathcal{N}_i^{\text{in}}|} \sum_{j\in\mathcal{N}_i^{\text{in}}} \boldsymbol{\eta}_j = \Big(\frac{1}{|\mathcal{N}_i^{\text{in}}|} \sum_{j\in\mathcal{N}_i^{\text{in}}} \boldsymbol{\theta}_j^k(0)\Big) - \boldsymbol{\theta}_m^k(0) \,, \quad (9)$$

where the equality is due to the fact that $\boldsymbol{\theta}_i^k(\infty) = \boldsymbol{\theta}_j^k(\infty)$ for all i, j regardless of the hypothesis [cf. Fact 1 and Proposition 1]. Now, suppose that the trustworthy agents are *initialized* using a distribution of the same mean $\bar{\boldsymbol{\theta}}$ while the attackers' initial values follow a distribution of the mean $\bar{\boldsymbol{x}}$, such that $\bar{\boldsymbol{x}} \neq \bar{\boldsymbol{\theta}}$. From (9), it is obvious that $\mathcal{D}^i = 0$ when there is *no attacker* in the neighborhood $\mathcal{N}_i^{\text{in}}$; while $\mathcal{D}^i = \Omega\left(|\mathbf{1}^\top(\bar{\boldsymbol{x}} - \bar{\boldsymbol{\theta}})|\right) \neq 0$ when there is *at least one attacker* in the neighborhood $\mathcal{N}_i^{\text{in}}$.

Formally, to characterize the detection performance, we need the following assumptions on the statistics of the initial values for trustworthy agents and the attackers:

H6. *A trustworthy agent $i \in N$ is initialized by $\boldsymbol{\theta}_i^k(0)$, which is a sub-Gaussian random vector with mean $\bar{\boldsymbol{\theta}}$ and covariance $\sigma_\theta^2 \boldsymbol{I}$.*

H7. *An attacker $j \in A$ is initialized by $\boldsymbol{\theta}_j^k(0)$ defined in* (4)*, which is a sub-Gaussian random vector with mean $\bar{\boldsymbol{x}}$, and covariance $\sigma_x^2 \boldsymbol{I}$, and $\boldsymbol{z}_j^k(0)$ is a sub-Gaussian random vector with zero mean, and covariance $\sigma_z^2 \boldsymbol{I}$.*

Under the assumptions above, the following can be established:

Proposition 2. *Define* $\mu_i = \frac{|N \cap \mathcal{N}_i^{\text{in}}|}{|\mathcal{N}_i^{\text{in}}|} \mathbf{1}^\top (\bar{\boldsymbol{x}} - \bar{\boldsymbol{\theta}})$ *and*

$$\sigma_i^2 = \left(\frac{|\mathcal{N}_i^{\text{in}}|^2 - 2|\mathcal{N}_i^{\text{in}}| + |A \cap \mathcal{N}_i^{\text{in}}|}{|\mathcal{N}_i^{\text{in}}|^2} \right) \sigma_z^2 + \frac{|N \cap \mathcal{N}_i^{\text{in}}|^2}{|\mathcal{N}_i^{\text{in}}|^2} (\sigma_x^2 + \sigma_\theta^2) \, .$$

Under the stated assumptions, we have

$$\mathrm{P}(\mathcal{D}^i > \delta_I \mid \mathcal{H}_0^i) \leq 2|\mathcal{N}_i^{\text{in}}| \exp\left(-Kd \frac{\delta_I^2}{2\sigma_\theta^2 |\mathcal{N}_i^{\text{in}}| (|\mathcal{N}_i^{\text{in}}| - 1)} \right),$$

$$\mathrm{P}(\mathcal{D}^i > \delta_I \mid \mathcal{H}_1^i) \geq 1 - \exp\left(-Kd \frac{\max\{0, -\delta_I + |\mu_i|\}}{2\sigma_i^2} \right).$$

The proof of Proposition 2 is provided in Appendix 5.3. In general, observe that the performance improves when:

1) the number of instances K and dimension d accrued increases;
2) $|\mathbf{1}^\top (\bar{\boldsymbol{x}} - \bar{\boldsymbol{\theta}})|$ increases and the variances $\sigma_x, \sigma_z, \sigma_\theta$ decrease;
3) the number of trustworthy agents near agent i increases.

Apparently, as the number of attackers in the network increases, the detection performance will deteriorate.

Localization Task. Suppose that agent i detects the existence or presence of an attacker in his/her neighborhood. Our next focus is on the *localization task*, whose goal is to distinguish between:

$$\begin{aligned} \mathcal{H}_0^{ij} &\text{ — agent } j \text{ is not an attacker, } \textit{i.e.,}\ j \notin A, \\ \mathcal{H}_1^{ij} &\text{ — agent } j \text{ is an attacker, } \textit{i.e.,}\ j \in A, \end{aligned} \tag{10}$$

for all $j \in \mathcal{N}_i^{\text{in}}$. Similarly, we propose checking the metric:

$$\mathcal{L}^{ij} := \left| \mathbf{1}^\top \boldsymbol{\eta}_j \right| \underset{\mathcal{H}_0^{ij}}{\overset{\mathcal{H}_1^{ij}}{\lessgtr}} \epsilon_I, \quad j \in \mathcal{N}_i^{\text{in}} \tag{11}$$

for all $j \in \mathcal{N}_i^{\text{in}}$. Consider the same set of assumptions as before – H6, H7 – we observe that under $\mathcal{H}_0^{ij}$, the metric can be approximated as $\mathcal{L}^{ij} \approx |\mathbf{1}^\top \left(\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{x}} \right)| > 0$; while under $\mathcal{H}_1^{ij}$, the metric is approximately zero $\mathcal{L}^{ij} \approx 0$. The performance can be bounded as:

Proposition 3. *Under the stated assumptions, we have*

$$\mathrm{P}(\mathcal{L}^{ij} < \epsilon_I \mid \mathcal{H}_0^{ij}) < \exp\left(-Kd \frac{(\max\{0, -\epsilon_I + |\mathbf{1}^\top (\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{x}})|\})^2}{2(\sigma_x^2 + \sigma_\theta^2)} \right)$$

$$\mathrm{P}(\mathcal{L}^{ij} < \epsilon_I \mid \mathcal{H}_1^{ij}) \geq 1 - 2\exp\left(-Kd\epsilon_I^2 / (2\sigma_z^2) \right).$$

The proof of Proposition 3 is provided in Appendix 5.3. We observe that the localization performance improves under the same conditions 1) and 2) for that in detection.

4. SIMULATIONS AND CONCLUSIONS

In this section, we provide numerical experiments to evaluate the performance of the proposed detection and localization methods. We run several trails of the DPG, each of which contains K instances. We calculate the metrics in (8) and (11) based on the accumulated data and test the detection and localization performance. We consider a Manhattan topology with $n = 8$ normal nodes (c.f., (2)) and one stubborn node (c.f., (4)), as shown in Fig. 3 in [22]. For simplicity, we take an example of the least square problem; i.e.,

$$f^k(\boldsymbol{\theta}) = \textstyle\sum_{i=1}^n f_i^k(\boldsymbol{\theta}) = \sum_{i=1}^n \left| (\boldsymbol{a}_i^k)^T \boldsymbol{\theta} - y_i^k \right|^2, k = 1, ..., K.$$

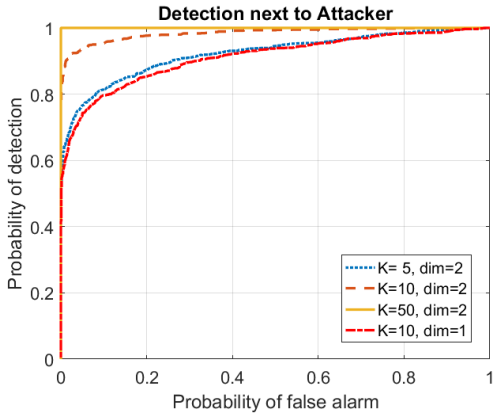

Fig. 1. ROCs temporal difference detection performance at the neighboring nodes of the attacker.

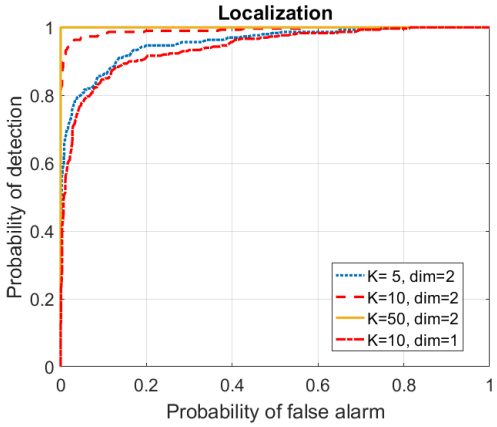

Fig. 2. ROCs temporal difference localization performance at the neighboring nodes of the attacker.

Herein, f_i^k can be seen as a utility function for instance k. We write the expected transition matrix as $\mathbb{E}[\boldsymbol{W}(t)] = \boldsymbol{I} - \frac{1}{2n}\boldsymbol{\Sigma} + \frac{\boldsymbol{P}+\boldsymbol{P}^\top}{2n}$. where $[\boldsymbol{P}]_{ij} = P_{ij}$ and $\boldsymbol{\Sigma}$ is defined as a diagonal matrix with $[\boldsymbol{\Sigma}]_{ii} = \sum_{j=1}^n (P_{ij} + P_{ji})$. Therefore, in our setting the randomized gossip-based decentralized protocol is run with probability $P_{ij} = 1/|\mathcal{N}_i^{\text{in}}|$ between node i and node j at each time t, and is terminated at $T_\infty = 2000$. The Monte Carlo simulation is run with 300 trials. At each trail under each instance, we set $\boldsymbol{x}^k \sim \mathcal{U}[-0.5, 0.5]^d$ and $z_j^k(t) \sim \mathcal{U}[-\hat{\lambda}^t, \hat{\lambda}^t]$, where λ is the second largest eigenvalue of $\mathbb{E}[\boldsymbol{W}(t)]$. Also, we change the specific functions $f_i^k(\boldsymbol{\theta})$ by randomly generating $\boldsymbol{a}_i^k \sim \mathcal{U}[0.5, 2.5]^d$, $(\boldsymbol{\theta}^\star)^k \sim \mathcal{U}[0, 1]^d$ and then let $y_i^k = (\boldsymbol{a}_i^k)^T (\boldsymbol{\theta}^\star)^k$ and pass $\boldsymbol{a}_i^k$ and y_i^k to the decentralized optimization algorithm with the initialization $\boldsymbol{\theta}^k(0) \sim \mathcal{U}[0, 1]^d$. In the simulation, we take $d = 2$

In Figure 1, we show curves of the attacker detection performance at the nodes next to an attacker. In the legend, K is the number of instances; dim means that among total d dimensions, how many of which is observed to calculate the metric. If $dim = 2$, we have exactly the same metric as (8) and (11); if $dim = 1$, we replace the vector $\mathbf{1}$ in (8) and (11) by the vector $[1, 0]$. We can see that, by employing the temporal difference method, the trustworthy agents next to an attacker can detect the network is under attack. The detection performance improves with K when dim is fixed, and improves with dim when K is fixed. When the attacker is next to the

trustworthy agent, we can detect the attacker when the product of K and dim is 100, but in general localization appears to be more demanding. In Figure 2, we show the localization ROC curves. Notice that the localization task is invoked only when one trustworthy agent has detected an attack in the neighborhood. In the simulations we assume that the neighborhood detection test was completed correctly, which means that the performance shown are optimistic. The numerical results show that the tests improve with K and d. The test is quite reliable when the product of K and dim is 100.

To conclude, in this paper we have studied how to detect the malicious node in a network running an instance of the DPG optimization algorithm. Considering the nature of the problem, we first prove that under a coordinated attack, the attackers can steer the network towards their desired optimum point. We therefore proposed a attack detection and a localization method and studied its performance analytically and by simulations. The efficacy of the method is demonstrated by numerical experiments. Future work include the extension to other distributed optimization algorithms, the analysis of other detection strategies and the Byzantine resilience of our detection methods.

5. APPENDIX

5.1. Proof of Proposition 1

Define the following product of matrices:

$$\boldsymbol{\Phi}(t,s) := \boldsymbol{W}(t)\boldsymbol{W}(t-1)\cdots\boldsymbol{W}(s),\ t \geq s\ , \tag{12}$$

and $\Phi_{ij}(t,s) := [\boldsymbol{\Phi}(t,s)]_{ij}$. For $i \in N$, we observe the inequality:

$$\begin{aligned}
&\|\boldsymbol{\theta}_i(t+1) - \boldsymbol{x}\| = \big\|\mathcal{P}_{\mathcal{C}}\big(\bar{\boldsymbol{\theta}}_i(t) - \gamma(t)\nabla f_i(\bar{\boldsymbol{\theta}}_i(t))\big) - \boldsymbol{x}\big\| \\
&\leq \big\|\bar{\boldsymbol{\theta}}_i(t) - \boldsymbol{x}\big\| + \gamma(t)\|\nabla f_i(\bar{\boldsymbol{\theta}}_i(t))\| \\
&\leq \Big\|\sum_{j\in N} W_{ij}(t)(\boldsymbol{\theta}_j(t) - \boldsymbol{x})\Big\| + \sum_{j\in A} W_{ij}(t)\|\boldsymbol{z}_j(t)\| + \gamma(t)M.
\end{aligned}$$

Here, the first inequality is due to the projection inequality. Using the fact that $\sum_{j\in A} W_{ij}(t) \leq 1$, the second last term can be bounded by $\bar{z}(t) := \max_{j\in A}\|\boldsymbol{z}_j(t)\|$.

Now define $B := (n-1)B_1 + B_2$, we can proceed with the recursion above to get:

$$\begin{aligned}
\|\boldsymbol{\theta}_i((k+1)B) - \boldsymbol{x}\| &\leq \|\sum_{j\in N}\Phi_{ij}((k+1)B-1, kB)(\boldsymbol{\theta}_j(kB) - \boldsymbol{x})\| \\
&+ \textstyle\sum_{\ell=kB}^{(k+1)B-1}\big(\gamma(\ell)M + \bar{z}(\ell)\big)\ .
\end{aligned} \tag{13}$$

Using the triangular inequality, it is easy to bound the first term on the right hand side above as:

$$\Big(\sum_{j\in N}\Phi_{ij}((k+1)B-1, kB)\Big)\cdot\max_{j\in N}\big\|\boldsymbol{\theta}_j(kB) - \boldsymbol{x}\big\|.$$

Let us invoke the following lemma, whose proof can be found in Appendix 5.2:

Lemma 1. *Under H1, H3(a), H5. It holds that*

$$\max_{i\in N}\textstyle\sum_{j\in N}\phi_{ij}(t+B, t+1) \leq 1-\eta^B < 1,\ \forall\, t \geq 0\ . \tag{14}$$

Substituting the above into (13) and (14) yields the inequality:

$$\begin{aligned}
\max_{i\in N}\|\boldsymbol{\theta}_i((k+1)B) - \boldsymbol{x}\| \leq (1-\eta^B)\cdot\max_{i\in N}\big\|\boldsymbol{\theta}_i(kB) - \boldsymbol{x}\big\| \\
+ \textstyle\sum_{\ell=kB}^{(k+1)B-1}\big(\gamma(\ell)M + \bar{z}(\ell)\big),
\end{aligned}$$

for all $k \geq 1$. As $B < \infty$, the latter term vanishes as $k \to \infty$. Combining these observations and applying Corollary 3 in [23] implies that $\max_{i\in N}\big\|\boldsymbol{\theta}_i(kB) - \boldsymbol{x}\big\| \to 0$. Consequently, (5) holds since for all $s \in [1, B-1]$, $\boldsymbol{\theta}_i(kB+s) - \boldsymbol{\theta}_i(kB)$ can also be bounded by a vanishingly small quantity.

5.2. Proof of Lemma 1

Define $\boldsymbol{D}(t) := [W_{ij}(t)]_{i,j\in N}$ and $\boldsymbol{\Phi}_D(t,s) := [\Phi_{ij}(t,s)]_{i,j\in N}$, *i.e.,* the lower right sub-block of $\boldsymbol{D}(t), \boldsymbol{\Phi}(t,s)$. It can be shown that

$$\boldsymbol{\Phi}_D(t,s) = \boldsymbol{D}(t)\boldsymbol{D}(t-1)\cdots\boldsymbol{D}(s),\ t \geq s\ . \tag{15}$$

We are interested in the matrix-vector product $\boldsymbol{\Phi}_D(t+B, t+1)\mathbf{1}$. Notice that under H1, H5(b), there exists $t^\star \in [t+1, t+B_2]$ such that:

$$[\boldsymbol{\Phi}_D(t^\star, t+1)\mathbf{1}]_{j^\star} \leq 1-\eta\ \text{ for some } j^\star \in N\ . \tag{16}$$

Since $t + B - t^\star \geq (n-1)B_1$, under H5(a) and using similar arguments as in the proof of Lemma 2 in [4], we can show

$$\Phi_{ij}(t+B, t^\star+1) \geq \eta^{t+B-t^\star},\ \forall\, i,j \in N\ . \tag{17}$$

Consequently, for all $i \in N$, we have:

$$\begin{aligned}
&[\boldsymbol{\Phi}_D(t+B, t+1)\mathbf{1}]_i = \sum_{\ell\in N}\Phi_{i\ell}(t+B, t^\star+1)[\boldsymbol{\Phi}_D(t^\star, t+1)\mathbf{1}]_\ell \\
&\leq 1 - \eta\cdot\Phi_{i,j^\star}(t+B, t^\star+1) \leq 1-\eta^B\ ,
\end{aligned}$$

where the first equality is obtained by simply expanding the matrix-matrix product; and the second inequality is due to (16). This concludes the proof of Lemma 1.

5.3. Proof of Propositions 2 and 3

The proof is similar to that in [22, Theorem 1]. It is easy to check that under $\mathcal{H}_0^i$, $\mathcal{D}_i$ is a zero mean r.v. with sub-Gaussian parameter $\sigma_\theta^2(|\mathcal{N}_i^{\text{in}}|-1)/(Kd|\mathcal{N}_i^{\text{in}}|)$. Applying the union bound gives

$$\begin{aligned}
&\mathrm{P}(\mathcal{D}^i > \delta_I \mid \mathcal{H}_0^i) \\
&\leq |\mathcal{N}_i^{\text{in}}|\mathrm{P}\left(\left|\mathbf{1}^\top\left(\boldsymbol{\eta}_m - \sum_{j\in\mathcal{N}_i^{\text{in}}}\frac{\boldsymbol{\eta}_j}{|\mathcal{N}_i^{\text{in}}|}\right)\right| > \frac{\delta_I}{|\mathcal{N}_i^{\text{in}}|} \mid \mathcal{H}_0^i\right), \exists m \in \mathcal{N}_i^{\text{in}},
\end{aligned}$$

By using Hoeffding's inequality [24] on the above, we obtain the desired result for the false alarm rate.

On the other hand, under $\mathcal{H}_1^i$, we have $\mathbf{1}^\top(\boldsymbol{\eta}_m - \frac{1}{|\mathcal{N}_i^{\text{in}}|}\sum_{j\in\mathcal{N}_i^{\text{in}}}\boldsymbol{\eta}_j)$ being a r.v. with mean μ_i and sub-Gaussian parameter $\sigma_i^2/(Kd)$ for $m \in A \cap \mathcal{N}_i^{\text{in}}$. Then,

$$\begin{aligned}
&\mathrm{P}(\mathcal{D}^i < \delta_I \mid \mathcal{H}_1^i) \leq \mathrm{P}\left(\left|\boldsymbol{e}_j^\top\left(\boldsymbol{\eta}_m - \sum_{j\in\mathcal{N}_i^{\text{in}}}\frac{\boldsymbol{\eta}_j}{|\mathcal{N}_i^{\text{in}}|}\right)\right| < \delta_I \mid \mathcal{H}_1^i\right), \forall m \in \mathcal{N}_i^{\text{in}}, \\
&\leq \mathrm{P}\left(\underbrace{\boldsymbol{e}_j^\top\left(\boldsymbol{\eta}_m - \sum_{j\in\mathcal{N}_i^{\text{in}}}\frac{\boldsymbol{\eta}_j}{|\mathcal{N}_i^{\text{in}}|}\right) - \mu_i}_{\hat{\eta}} > -\delta_I + |\mu_i| \mid \mathcal{H}_1^i\right), \forall m \in \mathcal{N}_i^{\text{in}}.
\end{aligned}$$

Apparently, $\hat{\eta}$ is a sub-Gaussian r.v. with mean zero and parameter $\sigma_i^2/(Kd)$. By using Hoeffding's inequality we have the desired lower bound for the detected rate.

For the localization task, we have $\mathbf{1}^\top\boldsymbol{\eta}_j$ being a zero mean r.v. with sub-Gaussian parameter $\sigma_z^2/(Kd)$ under $\mathcal{H}_1^{ij}$, and being a sub-Gaussian r.v. with mean $\mathbf{1}^\top(\bar{\boldsymbol{x}} - \bar{\boldsymbol{\theta}})$ and variance $(\sigma_x^2 + \sigma_\theta^2)/(Kd)$ under $\mathcal{H}_0^{ij}$. Then, Hoeffding's inequality gives the desired bounds for localization performance. This completes the proofs.

6. REFERENCES

[1] J. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Dept. of Electrical Engineering and Computer Science, M.I.T., Boston, MA, 1984.

[2] J. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Trans. Autom. Control*, vol. 57, no. 3, pp. 592–606, March 2012.

[3] A. H. Sayed, *Adaptation, Learning, and Optimization over Networks*. Foundations and Trends in Machine Learning, 2014, vol. 7.

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

[5] S. Ram, A. Nedić, and V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *Journal of Optimization Theory and Applications*, vol. 147, no. 3, pp. 516–545, 2010.

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

[7] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip Algorithms for Distributed Signal Processing," *Proc. IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.

[8] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, ser. IPSN '04. New York, NY, USA: ACM, 2004, pp. 20–27. [Online]. Available: http://doi.acm.org/10.1145/984622.984626

[9] D. Jakovetic, J. Xavier, and J. M. F. Moura, "Fast distributed gradient methods," *IEEE Trans. Autom. Control*, vol. 59, no. 5, pp. 1131–1146, May 2014.

[10] E. Wei and A. Ozdaglar, "On the o(1/k) convergence of asynchronous distributed alternating direction method of multipliers," *CoRR*, 2013.

[11] Y. Yang, G. Scutari, D. P. Palomar, and M. Pesavento, "A parallel stochastic approximation method for nonconvex multi-agent optimization problems," *CoRR*, vol. abs/1410.5076, Oct 2014.

[12] T.-H. Chang, A. Nedic, and A. Scaglione, "Distributed constrained optimization by consensus-based primal-dual perturbation method," *IEEE Trans. Autom. Control*, vol. 59, no. 6, pp. 1524–1538, June 2014.

[13] H.-T. Wai, J. Lafond, A. Scaglione, and E. Moulines, "Decentralized frank-wolfe algorithm for convex and non-convex problems," *IEEE Transactions on Automatic Control*, 2017.

[14] P. Bianchi and J. Jakubowicz, "Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization," *IEEE Trans. Autom. Control*, vol. 58, no. 2, pp. 391–405, Feb 2013.

[15] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Sparse attack construction and state estimation in the smart grid: Centralized and distributed models," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1306–1318, 2013.

[16] O. Vuković and G. Dán, "Security of fully distributed power system state estimation: Detection and mitigation of data integrity attacks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 7, pp. 1500–1508, July 2014.

[17] X. Liu, Z. Bao, D. Lu, and Z. Li, "Modeling of local false data injection attacks with reduced network information," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1686–1696, 2015.

[18] S. Sundaram and B. Gharesifard, "Consensus-based distributed optimization with malicious nodes," in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2015, pp. 244–249.

[19] L. Su and N. Vaidya, "Byzantine multi-agent optimization: Part II," *CoRR*, July 2015.

[20] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, Sep. 2002.

[21] S. Zhu, S. Setia, and S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks," in *Proc CCS '03*, 2003, pp. 62–72.

[22] R. Gentz, S. X. Wu, H. T. Wai, A. Scaglione, and A. Leshem, "Data injection attacks in randomized gossiping," *IEEE Transactions on Signal and Information Processing over Networks*, vol. PP, no. 99, pp. 1–1, 2016.

[23] B. P. Polyak, *Introduction to Optimization*. Optimization Software, Inc., 1987.

[24] P. Massart, *Concentration Inequalities and Model Selection*. Springer, 2003.