Spoof Resilient Coordination in Distributed and Robust Robotic Networks

Venkatraman Renganathan, Student Member, IEEE, Kaveh Fathian, Student Member, IEEE, Sleiman Safaoui, Student Member, IEEE, and Tyler Summers, Member, IEEE

Abstract—As cyber-physical networks become increasingly equipped with embedded capabilities, they are made vulnerable to malicious attacks with the increased number of access points available to attackers. A particularly pernicious attack is spoofing, in which a malicious agent spawns multiple identities and can compromise otherwise attack-resilient algorithms that rely on assumed network robustness structures. We generalize a class of resilient consensus strategies, known as Weighted Mean Subsequence-Reduced (W-MSR) consensus, to provide spoof resilience by incorporating a physical layer authentication. By comparing the physical fingerprints of received signals, legitimate agents can identify and isolate malicious agents that attempt spoofing attacks. In stochastic settings where fingerprint signals are noisy, we quantify worst-case misclassification probability using distributionally robust Chebyshev bounds computed via semidefinite programming. Numerical simulations and experimental results illustrate the effectiveness of the proposed methods. Our framework is applicable to a variety of problems involving multi-robot systems coordinating via wireless communication.

Index Terms—Spoof Attack, Network Resiliency, Robot Coordination, Graph Robustness.

SUPPLEMENTARY MATERIAL

Video of the experimental results and simulation is available at https://youtu.be/dcd0EexNnzE

I. INTRODUCTION

As cyber-physical networks become increasingly equipped with embedded sensing, communication, computation, and actuation capabilities, they are made vulnerable to malicious attacks by increasing the number of access points available for attackers. A large and growing literature has emerged on security, resilience, and robustness of the cyber-physical systems in the presence of non-cooperative and adversarial agents [1][2][3][4]. Malicious agents under a distributed environment might gain undesirable advantage by influencing neighboring legitimate agents in the network. One such pernicious attack is spoofing, in which a malicious agent spawns multiple non-existent identities or impersonates existing legitimate agents.

Spoofing is not just an abstract concern; successful attacks have been realized in several critical networks, such as civilian GPS [6], global navigation satellite systems [7], anti-lock braking systems [8], and others.

Autonomous multi robot systems are a rapidly emerging type of cyber-physical network in which many tasks, including coverage, distributed estimation, cooperative manipulation, and formation control, utilize distributed consensus strategies, known as Weighted Mean Subsequence Reduced (W-MSR) consensus, to provide spoof resilience by incorporating a physical fingerprint analysis of signals received via wireless communication. This material is based on work supported by the National Science Foundation under Grant CNS-1561627.

1Also known as a “Sybil” attack [5].

Contributions: The present paper is a significant extension of our preliminary work in [23]. We propose an approach for spoof resilient coordination in cyber-physical and multi-robot networks. We generalize a class of resilient consensus strategies, known as Weighted Mean Subsequence Reduced (W-MSR) consensus, to provide spoof resilience by incorporating a physical fingerprint analysis of signals received from neighboring agents. Performing physical layer authen-
tication by comparing the physical fingerprints of received signals, legitimate agents can detect and isolate malicious agents that attempt spoofing attacks. Our algorithm achieves resilient consensus despite an arbitrary number of spoofed agents in the network. In stochastic settings where fingerprint signals are noisy, we quantify worst-case misclassification probability using distributionally robust Chebyshev bounds computed via semidefinite programming, using techniques in (24). Numerical simulations and experimental results using Sphero rolling robots swarms illustrate the effectiveness of the proposed methods. Our framework is applicable to a variety of problems involving multi-robot systems coordinating via wireless communication, including coverage, distributed estimation, and formation control.

The rest of the paper is organized as follows. Section II formulates a model for spoofing attacks in multi-robot networks and presents the attack detection technique using a physical fingerprint analysis. Section III proposes a generalized spoof resilient W-MSR algorithm by suitably modifying an existing standard W-MSR algorithm. A problem of quantifying misclassification probability is addressed in Section IV where a semidefinite programming formulation is proposed to arrive at distributionally robust Chebyshev bounds. Numerical simulations results are then presented in Section V. Section VI explains the experimental results obtained by implementing the proposed algorithms on a Sphero rolling robot swarm. Finally, Section VII summarizes the results and discusses future work directions.

II. Spoofing Attacks In Multi-Robot Networks

A. Spoofing Attack - Network Model

Figure 1 illustrates an example of a spoofing attack we aim to address. We model the network with an undirected graph \( G \) comprising a node set \( \mathcal{V} \) representing \( m \) agents and edge set \( \mathcal{E}[t] \subset \mathcal{V} \times \mathcal{V} \) representing a set of (possibly time-varying) communication links amongst the agents. The node set is partitioned into two disjoint subsets \( \mathcal{V} = S_I \cup S_R \). The set \( S_I \) represents the set of legitimate agents. A malicious agent attempts to disrupt the network by communicating subversive information to neighboring agents and may in addition attempt to perform a spoofing attack by creating multiple non-existent identities. Thus, the set of adversaries \( S_a \) is composed of both malicious and spoofed agents, so that \( S_a = S_m \cup S_s \), where \( S_m \) denotes malicious agents and \( S_s \) denotes the agents spoofed by \( S_m \). An upper bound of \( F \) number of malicious agents is assumed, whereas an arbitrary number of agents could be spoofed.

B. Consensus Dynamics - Update Model

We associate with each node \( i \in \mathcal{V} \), a state \( x_i[t] \in \mathbb{R} \) at time \( t \in \mathbb{Z}_{\geq 0} \). The state may represent a position or some quantity to be estimated or optimized, depending on the application context. In order to achieve some objective, the nodes interact synchronously by exchanging their state value with neighbors in the network (25, 26, 27). Let the set of inclusive neighbors be defined as \( \mathcal{J}_i[t] = \mathcal{N}_i[t] \cup \{i\} \), where \( \mathcal{N}_i[t] = \{ j \in \mathcal{V} : (j, i) \in \mathcal{E}[t] \} \) is the neighbor set of agent \( i \) at time \( t \), whose states are available to agent \( i \) via communication links. Each legitimate node updates its own state over time based on its current state and the state of neighboring agents according to a prescribed rule of the form

\[
x_i[t + 1] = f_i(x_j[t]), \quad j \in \mathcal{J}_i[t], \quad i \in S_I.
\]

The degree of \( i \) is denoted as \( \delta_i[t] = |\mathcal{N}_i[t]| \), and every node is assumed to have access to its own state at time \( t \).

Definition 1. A node \( i \in S \) is said to be malicious if it sends \( x_i[t] \) to all of its neighbors at each time-step, but applies some other function \( f'_i(\cdot) \) at some time-step (13).

Resilient consensus algorithms developed by (28) specify a nonlinear function \( f_i(.) \) that updates the states by suitably modifying which agents in the neighbor set, (including \( i \) \), \( \mathcal{J}_i[t] \) are included in the update to provide resilience to malicious agents.

C. Attack Detection Using Physical Fingerprint Analysis

We imagine a scenario in which the agents in the network communicate amongst themselves using a wireless communication protocol. We assume a complex multi-path environment where a transmitted signal is scattered off of walls and objects; manifesting themselves as measurable peaks in fingerprints and thereby contributing significantly to its uniqueness. Physical properties of the received wireless signal profiles are leveraged to detect the spoofing attack. The physical fingerprint of an agent \( j \) received by agent \( i \) at time \( t \) is modeled by a \( p \)-dimensional feature vector, \( \mathcal{F}_i[t] \in \mathbb{R}^p \) containing physical signal properties such as angle-of-arrival, time-of-arrival and other features. These features can be measured using a Synthetic Aperture Radar (SAR) as suggested by authors in (22). The received signals from SAR can be processed using well-studied signal processing algorithm called MUSIC to generate spatial fingerprint corresponding to each neighboring agent. Now, these spatial fingerprints can be used for discriminating between the signals received from two distinct agents as mentioned in (6). Since multiple spoofed agents may be generated.
by a single distinct malicious agent, the physical fingerprints associated with each of them will be similar. Of course, due to the random nature of wireless communication, there may be noise associated with the fingerprints of received signals. This situation can be modeled by associating a probability distribution with received signal fingerprints.

The neighbor set $\mathcal{N}_i$ of agent $i$ includes the set of agents which can transmit signals to agent $i$. Based on the received signal fingerprints of pairs of neighboring agents, we define a similarity metric

$$\gamma_{ij} = \frac{1}{1 + \|F_i^j - F_i^k\|}, \quad j, k \in \mathcal{N}_i,$$  

(2)

which quantifies how similar the fingerprint of neighboring agent $j$ is to that of neighboring agent $k$, as received by agent $i$. The authors in (22) deal with a similar setting for a coverage control problem and our development is inspired from their approach. Agent $i$ computes these similarity metrics for each neighbor pair. From these similarity metrics, a confidence weight $\alpha_{ij} \in [0, 1]$ can be associated with neighboring agent, which should be close to 1 for legitimate neighbors and close to 0 for spoofed and spoofing neighbors. For example, in a deterministic setting,

$$\gamma_{ij} = 1 \Rightarrow \alpha_{ij} = 0, \quad \alpha_{ik} = 0,$$
$$\gamma_{ij} < 1 \Rightarrow \alpha_{ij} = 1, \quad \alpha_{ik} = 1,$$  

(3)

i.e., the confidence weights for neighbors $j$ and $k$ are 0 if the neighbor $j$ has the same fingerprint as neighbor $k$, and 1 otherwise. In general, we can write

$$\alpha_{ij} = \prod_{j, k \in \mathcal{N}_i, j \neq k} (1 - \gamma_{ij}).$$  

(4)

In a stochastic setting, we define a spoof detection threshold $\omega \in [0, 1]$. If the likelihood that the physical fingerprints of two neighbors are different is below the threshold, the neighbors are classified as spoofed or spoofing agents, and otherwise they are classified as legitimate; for example,

$$g(\alpha_{ij}) \leq \omega \Rightarrow j \text{ is spoofed or spoofing},$$
$$g(\alpha_{ij}) > \omega \Rightarrow j \text{ is legitimate}.$$  

(5)

where $g(\cdot)$ is a prescribed detection function. In the stochastic settings, the $g(\cdot)$ and $\omega$ could be selected based on an assumed model for the probability distributions of the fingerprints and associated bounds on misclassification probability. To recover the deterministic case, we can set $\omega = 0$ and $g(x) = x$.

D. Example of a Physical Fingerprint Model

A specific example of a particular physical fingerprint model is discussed in (22). The fingerprint is modeled by a directional signal strength profile that depends on wireless signal wavelengths, distances and relative angles between directional antennae, multiple possible signal paths, and random channel properties with additive Gaussian noise. Based on this stochastic channel model, similarity and confidence metrics can be explicitly defined, and quantitative bounds can be obtained on the expectation that received signals are coming from spoofed agents. Such models could be used to define spoof resilient algorithms tailored to the specific communication model and perform analyses of probabilistic algorithm properties. Here we focus mainly on deterministic detection settings and stochastic settings with simple thresholding. Computing bounds associated with such specific probabilistic fingerprint models is described in Section IV.

III. DESIGN OF A SPOOF RESILIENT COORDINATION ALGORITHM

In this section, we describe a coordination algorithm that is resilient to anonymous malicious agents who share adversarial state values and may also attempt to spoof non-existent agent that also share adversarial state value. Since malicious agents do not all necessarily attempt to spoof, we build upon recent work on resilient consensus algorithms that do not handle spoofing. These algorithms achieve resiliency by effectively designing and exploiting redundancy in the communication graph. We now review these resilient graph properties and an existing resilient consensus algorithm called Weighted Mean-Subsequence-Reduced (W-MSR) as described in (13). We subsequently present our spoof resilient adaptation of W-MSR. We assume throughout that there are at most $F$ malicious agents but that there may be an arbitrary number of spoofed agents.

Definition 2. A set $S$ is $r$-reachable, if it contains a node that has at least $r$ neighbors outside of $S$. The parameter $r$ quantifies the redundancy of information flow from nodes outside of $S$ to some node inside $S$. Intuitively, the $r$-reachability property captures the notion that some node inside the set is influenced by a sufficiently large number of nodes from outside the set.

Definition 3. A graph $G = (\mathcal{V}, \mathcal{E})$ on $n$ nodes is said to be $r$-robust, with $r \in \mathbb{Z}_{\geq 0}$, if for every pair of disjoint nonempty subsets of $\mathcal{V}$, at least one of the subsets is $r$-reachable.

Definition 4. Given a graph $G$ and a nonempty subset of nodes $S$, we say that $G$ is an $(r,s)$ - reachable set, if there are at least $s$ nodes in $S$, each of which has at least $r$ neighbors outside of $S$, where $r, s \in \mathbb{Z}_{\geq 0}$. i.e., given $X_S = \{i \in S : |\mathcal{V}\setminus S| \geq r\}$, then $|X_S| \geq s$.

Definition 5. A graph $G = (\mathcal{V}, \mathcal{E})$ on $n$ nodes ($n \geq 2$) is $(r,s)$-robust, for nonnegative integers $r \in \mathbb{Z}_{\geq 0}$, $1 \leq s \leq n$, if for every pair of nonempty, disjoint subsets $S_1$ and $S_2$ of $\mathcal{V}$ such that $S_1$ is $(r, s, 1)$-reachable and $S_2$ is $(r, s, 2)$-reachable with $s_{r,1}$ and $s_{r,2}$ maximal (i.e., $s_{r,k} = |X_{S_k}|$ where $X_{S_k} = \{i \in S_k : |\mathcal{V}\setminus S_k| \geq r\}$ for $k = \{1, 2\}$), then at least one of the following hold:

1) $s_{r,1} = |S_1|$  
2) $s_{r,2} = |S_2|$  
3) $s_{r,1} + s_{r,2} \geq s$.

The $(r,s)$-robustness property introduces information redundancy by specifying a minimum number of nodes that are sufficiently influenced from outside of their set. Note that $(r,s)$-robustness is a strict generalization of $r$-robustness.
A. Resilient Asymptotic Consensus

Let \( x_M[t] \) and \( x_m[t] \) denote the maximum and minimum values of the legitimate nodes at time \( t \), respectively. Authors in \(^{[16]}\) explained that the legitimate agents in the network are said to achieve resilient asymptotic consensus in the presence of a particular threat model if for any initial conditions it holds

- \( \exists \ L \in \mathbb{R} \) such that \( \lim_{t \to \infty} x_i[t] = L, \forall i \in S_l \)
- the interval \([x_m[0], x_M[0]]\) is an invariant set (i.e., the legitimate values remain in the interval \( \forall t \))

Resilient asymptotic consensus has three important properties as explained in \(^{[14]}\). First, the legitimate nodes must reach asymptotic consensus despite the presence of some misbehaving nodes given a particular threat model and scope of threat (e.g., at most \( F \) malicious agents). This is a condition on agreement. Additionally, it is required that the interval containing the initial values of the legitimate nodes is an invariant set for the legitimate nodes; this is a safety condition, where the interval \([x_m[0], x_M[0]]\) is known to be safe. The agreement and safety conditions, when combined, imply a third condition on validity: the converged consensus value lies within the range of initial values of the legitimate nodes.

B. The Weighted Mean-Subsequence-Reduced (W-MSR) Algorithm

We now review a class of resilient consensus algorithms described in \(^{[13]}\) that utilize an update rule called as Weighted Mean-Subsequence-Reduced (W-MSR). At every time \( t \), each legitimate node \( i \) obtains the values of other nodes in its neighborhood. Since there are at most \( F \) total malicious nodes in the network, some of node \( i \)'s neighbors may misbehave; however, node \( i \) is unsure of which neighbors may be compromised. To ensure that node \( i \) updates its state in a safe manner, we consider a protocol where each node removes the extreme values with respect to its own value. Specifically, the W-MSR algorithm comprises the following steps:

1) At each time \( t \), each legitimate node \( i \in S_l \) obtains the state values of its neighbors, and forms a sorted list.

2) If there are less than \( F \) values strictly larger than its own value, \( x_i[t] \), then legitimate node \( i \) removes all values that are strictly larger than its own. Otherwise, it removes the largest \( F \) values in the sorted list (breaking ties arbitrarily). Likewise, if there are less than \( F \) values strictly smaller than its own value, then node \( i \) removes all values that are strictly smaller than its own. Otherwise, it removes precisely the smallest \( F \) values.

3) Let \( \mathcal{R}_i[t] \) denote the set of nodes whose values were removed by legitimate node \( i \) in step 2 at time \( t \). Each legitimate node \( i \) applies the update

\[
x_i[t+1] = \sum_{j \in \mathcal{J}_i[t] \setminus \mathcal{R}_i[t]} w_{ij}[t] x_j[t]
\]

where \( w_{ij}[t] \) is the weight associated with node \( j \)'s value by node \( i \) at time step \( t \). The weights are chosen to satisfy the following conditions:

\(^2\)In this case, a simple choice for the weights \(^{[12]}\) is to let \( w_{ij}[t] = 1/(1 + d_i[t] - |\mathcal{R}_i[t]|) \), for \( j \in \mathcal{J}_i[t] \setminus \mathcal{R}_i[t] \)

1) \( w_{ij}[t] = 0 \) whenever \( j \notin \mathcal{J}_i[t], i \in S_l, t \in \mathbb{Z}_{\geq 0} \)
2) there exists a constant \( \beta \in \mathbb{R}, 0 < \beta < 1 \) such that \( w_{ij}[t] \geq \beta, \forall j \in \mathcal{J}_i[t], i \in S_l, t \in \mathbb{Z}_{\geq 0} \)
3) \( \sum_{j=1}^{n} w_{ij}[t] = 1, \forall i \in S_l, t \in \mathbb{Z}_{\geq 0} \)

A network being \((F+1, F+1)\)-robust is a necessary and sufficient condition for the normal nodes to achieve consensus when no more than \( F \) total malicious nodes are present in the entire network \(^{[29]}\). Then if the graph is \((F+1, F+1)\)-robust, under the update protocol specified in equation \(^{(6)}\), the legitimate agents in network are guaranteed to achieve resilient asymptotic consensus \(^{[13]}\) despite the presence of at most \( F \) malicious agents, but assuming that there are no spoofed agents.

**Theorem 1.** Consider an undirected network \( G = (\mathcal{V}, \mathcal{E}) \), where \( \mathcal{V} \) represents the set of agents and \( \mathcal{E} \) represents the set of communication links between them. Suppose that the network is \((F+1, F+1)\)-robust, where \( F \) is the total number of malicious agents in the network. Suppose a malicious agent who attempts to spoof is connected to at least one legitimate agent with an intention to influence the network. Then, a spoofing attack by the malicious node can compromise the \((F+1, F+1)\)-robust property of the graph resulting in the failure of the W-MSR protocol to achieve resilient asymptotic consensus.

**Proof.** Consider a malicious node spoofing one non-existent spoofed identity which is connected to at least one other legitimate node with an intention to disrupt the network. We indicate by \( G' \), the new graph that is obtained by addition of the new node to the existing graph \( G \). Here, we refer to the Theorem 5.6 in \(^{[30]}\), which states that addition of nodes to an existing robust graph \( G \) will result in graphs \( G \) and \( G' \) having at most the same robustness property. That is, node addition will not improve graph robustness instead at most preserves the graph robustness. Since the new node being added is the spoofed entity, which is malicious in nature, we exceed the assumed upper bound on the number of malicious nodes in the network and hence the graph robustness property weakens in this specific attack setting. Thus, a spoofing attack by any such malicious node can compromise the \((F+1, F+1)\)-robust property of the graph resulting in the failure of the W-MSR protocol to achieve resilient asymptotic consensus. \( \square \)

C. Spoof Resilient W-MSR Algorithm

A spoofing attack is capable of compromising the \((F+1, F+1)\)-robust graph robustness property and W-MSR algorithm above, and hence the network resiliency. Our spoof resilient adaptation of the W-MSR algorithm here is summarized in Algorithm \(^{[1]}\). Based on a pairwise comparison of physical fingerprints of signals received from neighboring agents and associated confidence weights, spoofed agents in the network are identified. Achieving resiliency then involves removing the identified spoofed and spoofing agents from the state update if their confidence weight is at most equal to the spoofing threshold \( \omega \). Thus in a stochastic setting, a spoofing threshold can be employed as explained in the spoof resilient W-MSR extension presented in Algorithm \(^{[1]}\).
Lemma 1. In a deterministic setting, let us consider an undirected network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the set of agents and $\mathcal{E}$ represents the set of communication links between them. Suppose the network is $(F+1, F+1)$-robust, assuming an upper bound of $F$ total malicious agents in the network, some of which may spoof. Then the network achieves resilient asymptotic consensus under Algorithm 7 in the presence of any spoofing attack.

Proof. In a deterministic setting, the physical fingerprints of signals of spoofed agents are identical to that of the spoofing agent. So any spoofed and spoofing agents are exactly detected and removed from the network state updates in lines 17-20 of the proposed Algorithm 1. Moreover, since the initial graph is $(F+1, F+1)$-robust, the use of the W-MSR protocol for the state updates guarantees resilient asymptotic consensus in the presence of up to $F$ malicious nodes who do not spoof but may behave in other adversarial ways. Thus, the overall protocol is resilient to both an arbitrary number of spoofed agents and up to $F$ non-spoofing malicious agents in the network.

D. Choosing a Spoofing Threshold

Since robots have physical extent, there is a non-zero minimum distance between the sensors or receivers located in each robot that are used for discriminating received signals. We assume that this minimum distance translates to a corresponding minimum distance $\mathcal{F}_{\text{min}}$ in feature vector space, where the pairwise comparisons are made, so that

\[
\|\mathcal{F}_i^t - \mathcal{F}_k^t\| \geq \mathcal{F}_{\text{min}}, \quad \forall j, k \in \mathcal{N}_i, j \neq k. \tag{7}
\]

This suggests a threshold for robustly classifying neighboring agents whose fingerprints satisfy this bound as malicious or spoofed. Specifically, the similarity metrics and confidence weights for legitimate neighbors then satisfy

\[
\gamma_{i}^{jk} \leq \frac{1}{\mathcal{F}_{\text{min}}} \implies \alpha_{i}^{j} \geq \left(1 - \frac{1}{\mathcal{F}_{\text{min}}} \right)^{|\mathcal{N}_i|}. \tag{8}
\]

It follows that the threshold

\[
\omega = \left(1 - \frac{1}{\mathcal{F}_{\text{min}}} \right)^{|\mathcal{N}_i|} \tag{9}
\]

correctly discriminates between legitimate and malicious or spoofed neighbors, assuming that the fingerprints satisfy (7). However, fingerprints of received signals may be stochastic and not easily bounded. In this setting there may inherently be a possibility of misclassifying a malicious neighbor as legitimate. In the next section, we explore the problem of quantifying misclassification probabilities in spoofing detection via physical fingerprint authentication.

IV. QUANTIFYING MISCLASSIFICATION PROBABILITIES IN SPOOFING DETECTION

In our proposed physical layer authentication technique, each legitimate robot associates a fingerprint with each wireless communication signal received from neighboring agents. This fingerprint is represented as a vector of signal parameters, such as received signal strength and relative bearing. Since communication channels are inherently noisy, the pairwise comparison performed in Algorithm 1 to classify neighbors as legitimate or malicious/spoofed may incorrectly classify a spoofed neighbor as legitimate or vice versa. By associating probability distributions with fingerprints of received signals, misclassification probabilities can be estimated based on assumptions about the distributions.

In practice, true fingerprint distributions are generally unknown. Based on experimental data with sensors or signal receivers, distributions parameters such as mean and covariance can be estimated. It is common to assume that distributions are Gaussian to maintain tractability of certain probabilistic computations. However, the emerging area of distributionally robust optimization has shown that it is often not necessary to make such strong and often unjustifiable assumptions (31). Here we adopt a distributionally robust approach to compute the worst case probabilities of misclassification over the set of fingerprint distributions with given mean and covariance. These worst-case estimates can be interpreted as generalized Chebyshev bounds (24), which extend classical moment-based Chebyshev inequalities to vector-valued random variables.

Algorithm 1 Spoof Resilient W-MSR (SR-W-MSR)

\begin{algorithm}
\begin{algorithmic}[1]
\Procedure{SR-W-MSR}{$\omega$}
\State \Comment{Input: spoofing threshold $\omega$, convergence threshold $\epsilon$, initial states $x[0]$, received signal fingerprints $\mathcal{F}_i^t$ for each agent at each time $t$}
\State $t \leftarrow 0$
\While{$\|x[t+1] - x[t]\| \geq \epsilon$}
\State $i \leftarrow 1$
\While{$i \leq |\mathcal{S}|$}
\For{$j \in \mathcal{N}_i$}
\State $\alpha_{i}^{j} \leftarrow 1$
\EndFor
\For{$k \in \mathcal{N}_i$}
\If{$j \neq k$}
\State $\gamma_{i}^{jk} = \frac{1}{1 + \|\mathcal{F}_i^t - \mathcal{F}_j^t\|}$
\EndIf
\State $\alpha_{i}^{j} \leftarrow \alpha_{i}^{j} \left(1 - \gamma_{i}^{jk} \right)$
\EndFor
\If{$\alpha_{i}^{j} \leq \omega$}
\Comment{Spoof attack is detected}
\State $z[t] \leftarrow \mathcal{R}[t] \cup \{j\}$
\State $x_i[t+1] \leftarrow \sum_{j \in \mathcal{J}_i[t]} z[t] w_{ij}[t] x_j[t]$
\Else
\Comment{Spoof attack is not detected}
\State $x_i[t+1] \leftarrow \sum_{j \in \mathcal{J}_i[t]} \mathcal{R}[t] w_{ij}[t] x_j[t]$
\EndIf
\EndWhile
\State $i \leftarrow i + 1$
\EndWhile
\EndProcedure
\end{algorithmic}
\end{algorithm}
Let there be \( n \) neighbors in total for a given signal receiving robot who performs pairwise fingerprint comparisons. The setup is illustrated in Figure 2. The set of neighbors consists of both legitimate and bad neighbors. The set of bad neighbors consists of both malicious neighbors and any arbitrary number of spoofed identities emulated by each malicious neighbor. Let us denote the set of legitimate neighbors by \( \mathcal{L} = \{1, 2, \ldots, n_l\} \), the set of malicious neighbors by \( \mathcal{M} = \{1, 2, \ldots, n_m\} \), and finally we collectively refer to the set of all spoofed entities by \( \mathcal{S} = \{1, 2, \ldots, n_s\} \). To be precise, \( n_s = \sum_{m \in \mathcal{M}} n^s_m \), \( n^s_m \) refers to the number of spoofed entities emulated by the malicious agent, \( m \in \mathcal{M} \). Let \( X_l \sim \mathbf{P}_L(\mu_L, \Sigma_L) \) and \( X_m \sim \mathbf{P}_M(\mu_M, \Sigma_M) \) denote the fingerprints and associated probability distributions of the neighbors received by a robot, where \( X_l \) corresponds to a legitimate robot and \( X_m \) to a malicious one. We assume that the true distributions \( \mathbf{P}_L \), and \( \mathbf{P}_M \) are unknown but that the means and covariance matrices \( \mu_L, \Sigma_L, \mu_M, \Sigma_M \) are known or can be estimated from received signal data or sensor hardware datasheets (with \( \mathbf{S}^p \) denoting the set of symmetric \( p \times p \) matrices).

If a malicious agent attempts a spoofing attack, then all fingerprints of the malicious agent and its spoofed identities are distributed according to \( \mathbf{P}_M \). That is, \( \mu_S = \mu_M \) and \( \Sigma_S = \Sigma_M \) respectively. We are interested in probabilistic events that will result in a legitimate neighboring robot being wrongly classified as a malicious robot or vice-versa. The robot which receives signals from its \( n \) neighbors performs at most \( \binom{n}{2} \) pairwise comparisons. Out of those pairwise comparisons, we enumerate the possible events that will result in misclassification. We now interpret \( F_{\text{min}} \) as a threshold used to make classification decisions but that may result in misclassification due to possibly unbounded fingerprint distributions. A misclassification event occurs when

1. a legitimate neighbor’s fingerprint realization is closer than \( F_{\text{min}} \) to any malicious neighbor’s fingerprint realization,
2. a legitimate neighbor’s fingerprint realization is closer than \( F_{\text{min}} \) to a spoofed fingerprint realization of any malicious neighbor,
3. a legitimate neighbor’s fingerprint realization is closer to a malicious neighbor’s fingerprint realization than the corresponding malicious neighbor’s realization is to one of its own spoofed realizations,
4. a legitimate neighbor’s fingerprint realization is closer to a spoofed realization than the corresponding malicious neighbor’s realization which emulated the spoofed entity.

To model the above scenarios for particular pairs or triples of legitimate, malicious, and spoofed agents \( l \in \mathcal{L}, m \in \mathcal{M}, s \in \mathcal{S} \), we define the events

\[
\begin{align*}
C_{1,l,m} & = \{ X_l, X_m : \|X_l - X_m\|^2 \leq F_{\text{min}} \}, \\
C_{2,s} & = \{ X_l, X_s^s : \|X_l - X_s^s\|^2 \leq F_{\text{min}} \}, \\
C_{3,l,m,s} & = \{ X_l, X_m, X_s^s : \|X_l - X_m\|^2 \leq \|X_m - X_s^s\|^2 \}, \\
C_{4,l,m,s} & = \{ X_l, X_m, X_s^s : \|X_l - X_m\|^2 \leq \|X_m - X_s^s\|^2 \},
\end{align*}
\]

and denote corresponding families of these events as

\[
\begin{align*}
C_1 & = \{ C_{1,l,m} : l \in \mathcal{L}, m \in \mathcal{M} \}, \\
C_2 & = \{ C_{2,s} : l \in \mathcal{L}, s \in \mathcal{S} \}, \\
C_3 & = \{ C_{3,l,m,s} : l \in \mathcal{L}, m \in \mathcal{M}, s \in \mathcal{S} \}, \\
C_4 & = \{ C_{4,l,m,s} : l \in \mathcal{L}, m \in \mathcal{M}, s \in \mathcal{S} \}.
\end{align*}
\]

Since the fingerprints are independent, the joint random variable denoted by \( X = [X_l, X_m, X_s] \) has mean and covariance

\[
\mu = \begin{pmatrix} \mu_L \\ \mu_M \\ \mu_S \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \Sigma_L & 0 & 0 \\ 0 & \Sigma_M & 0 \\ 0 & 0 & \Sigma_S \end{pmatrix}.
\]

For \( k = 1, \ldots, 4 \) and \( i = 1, \ldots, |C_k| \) respectively, the probability of misclassification event \( i \) is given by

\[
\sup \{ \mathbf{P}(X \in C_k^i) \} = 1 - \inf \{ \mathbf{P}(X \notin C_k^i) \},
\]

where the supremum and infimum are taken over the set of probability distributions on \( \mathbb{R}^p \) with the given mean and covariance. Utilizing generalized Chebyshev bounds from (24), the misclassification probabilities can be readily computed by solving a (convex) semidefinite programming problem. Though the events are generally independent within the same family, they are generally dependent between families. So, we address them separately here rather than jointly.

### A. Computing Misclassification Probability via Semidefinite Programming

The first step in computing the misclassification probability for a particular pair of legitimate and malicious agents, or
triple of legitimate, malicious, and spoofed agents, is to form the neighbor pair combination matrix $A_{C_k}$ corresponding to an event $i$ in the family $C_k$. Assume a total of $n = 3$ neighbors with $n_l = n_m = n_s = 1$ and thus, $X = [X_l, X_m, X_s]^T$. Now, for $i = 1, \ldots, |C_k|$, we have

$$A_{C_i} = \begin{pmatrix}
1 & -1 & 0 \\
-1 & 1 & 0 \\
0 & 0 & 0
\end{pmatrix} \otimes I_p,$$

$$A_{C_2} = \begin{pmatrix}
1 & 0 & -1 \\
0 & 0 & 0 \\
-1 & 0 & 1
\end{pmatrix} \otimes I_p,$$

$$A_{C_3} = \begin{pmatrix}
1 & 0 & -1 \\
0 & -1 & 1 \\
-1 & 1 & 0
\end{pmatrix} \otimes I_p,$$

$$A_{C_4} = \begin{pmatrix}
0 & -1 & 1 \\
-1 & 1 & 0
\end{pmatrix} \otimes I_p. \quad (14)$$

We now have the following result for computing distributionally robust misclassification probabilities using semidefinite programming.

**Theorem 2.** Consider the joint random fingerprint variable $X = [X_l, X_m, X_s]^T$ with mean $\mathbf{E}[X] = \mu$ and covariance $\mathbf{E}[XX^T] = \Sigma$. Now consider an event $i$ in the family $C_k$, $k \in \{1, 2, 3, 4\}$, with $A_{C_i}$ formed using (14). Let $\kappa = \begin{cases} \lambda \mathcal{F}_{\text{min}}, & \text{if } k = 1, 2 \\ 0, & \text{otherwise} \end{cases}$. The worst-case probability of misclassification

$$\sup \{ \mathbf{P}(X \in C_k) \}$$

over the set of all distributions of $X$ with mean $\mu$ and covariance $\Sigma$ is given by the optimal value of the following semidefinite program

$$\begin{array}{c}
\text{maximize} \\
\text{subject to} \\
\text{with variables} \end{array} \begin{array}{c}
\lambda \\
\text{tr}(A_{C_i}^T Z) \leq \kappa, \\
\begin{pmatrix}
\Sigma + \mu \mu^T & \mu \\
\mu^T & 1
\end{pmatrix} \geq 0,
\end{array}$$

with variables $\lambda \in \mathbb{R}$, $Z \in \mathbb{S}^p$, $z \in \mathbb{R}^p$.

**Proof.** Here we demonstrate a proof for a misclassification event $i$ from the family $C_3$. The proofs corresponding to events in the other families are entirely analogous. In this case the event involves one legitimate neighbor $X_l$, one malicious neighbor $X_m$ and one spoofed identity $X_s$. Thus, three signals are received by the robot, whose job is to classify the neighbors based on their signal properties. Let us denote the joint random variable as $X = [X_l, X_m, X_s]^T$. Let $A = (X_l - X_m)$, $B = (X_m - X_s)$. Now, the smallest probability of not misclassifying, namely $\inf \mathbf{P}(X \notin C_3)$, is given by

$$\inf \mathbf{P}(\|A\|^2 \geq \|B\|^2)$$

$$= \inf \mathbf{P}(A^T A - B^T B \geq 0)$$

$$= \inf \mathbf{P}(X^T \begin{pmatrix}
1 & -1 & 0 \\
0 & 1 & -1
\end{pmatrix} X \geq 0)$$

$$= \inf \mathbf{P}(X^T X \leq 0).$$

Then applying a generalized Chebyshev inequality (from the main result in Section 2 of (24)) and combining with (13) yields the result.\qed

**B. Illustrative Example**

![Fig. 3: Misclassification probability with respect to events in the family $C_3$ decreases as the minimum spatial distance $F_{\text{min}}$ parameter is increased.](image)

Suppose that the robots move in $\mathbb{R}^2$ and the receiving robot has sensors that allow discrimination of received signal strength and bearing, which are used to estimate the positions of neighboring robots, which then serve as the fingerprint estimate. The signal strength measurement in the radial direction toward the neighbors and the bearing measurement in the orthogonal direction are given positive variance values, corresponding to a highly noisy scenario. For illustrative purposes, suppose the receiving robot is located at origin and the neighbors are located randomly around it. The semidefinite program (15) was solved for varying threshold values $F_{\text{min}}$ to compute the worst-case misclassification probability. Correspondingly, increasing the $F_{\text{min}}$ parameter resulted in a decrease of the misclassification probability with respect to the events in the family $C_3$ as shown in Figure $3$. Specifically, this type of misclassification probability decreases rapidly with increasing separation distance.

Similar analogous semidefinite programming problems can be formulated for other events capturing misclassification probabilities. These estimates of misclassification probability can be sharpened further by making stronger assumptions on
the fingerprint distributions, e.g., symmetry or knowledge of higher order moments. In any case, although misclassifications may occasionally occur, signals will be received relatively frequently. The probability of misclassifications persisting over extended periods of time would rapidly decrease, and robots may be able to improve spoofing classifications by considering fingerprints over multiple time periods. Such analysis will be pursued in future work.

V. NUMERICAL SIMULATIONS

We now illustrate our spoof resilient W-MSR algorithm using the 7-node (2,2)-robust network with 6 legitimate agents, 1 malicious agent who spoofs 1 agent, as shown in Figure 4. The objective of the network is to form a hexagonal formation and remain in the safe region, which can be expressed by introducing a constant bias in the consensus update equations. Specifically, at every iteration the desired position of robot $i = 1, \ldots, 6$ is computed via

$$x_i[t + 1] = \sum_{j \in J_i \setminus \{i\}} w_{ij}[t] (x_j[t] - \bar{x}_j) + \bar{x}_i,$$

where $\bar{x}_i := [\sin(\theta_i), \cos(\theta_i)]^\top \in \mathbb{R}^2$, $\theta_i := \frac{2\pi(i-1)}{6}$, is a constant bias vector that is used to position each robot at a vertex of the hexagon. Note that due to the addition and subtraction of the same bias vector for each agent, by defining $\bar{x}_i := x_i - \bar{x}_i$ one can see from (16) that $\bar{x}_i$ has the same dynamics as of Eq. (16). The six legitimate agents were given random initial states as shown in Figure 5a. Malicious node 7 performs spoofing attack by spawning a spoofed identity called 8 and sends messages to all his neighbors using both identities. A constant bias of $\bar{x} = 5\text{cm}$ is added to malicious robot’s position in each dimension to obtain spoofed robot’s position. The goal of the malicious and spoofed robots is to move the legitimate robot formation from safe region to unsafe region. A spoofing attack is evaluated both in deterministic and stochastic settings. In the deterministic setting, all the physical fingerprints are obtained without any noise. As a result, the spoofed nodes are exactly identified and removed from the network using the spoof resilient W-MSR algorithm.

The rest of the section is organized as follows. Initially the failure of linear consensus protocols under malicious attacks is demonstrated in Subsection V-A. Then, in Subsection V-B the W-MSR algorithms is demonstrated to achieve resiliency against the malicious attack. The failure of the W-MSR algorithm under spoofing attack is presented in Subsection V-C. Implementation of the proposed spoof resilient extension of W-MSR algorithm is presented in Subsection V-D. Finally other additional observations are discussed in Subsection V-E.

A. Linear Consensus Protocol Fails with 1 Malicious Agent

Consider the network as shown in Figure 4 with spoofed agent not being present. When a linear consensus protocol is employed to obtain a robot formation, the malicious robot is successful in pulling the formation from safe region to the unsafe region as shown in Figure 5a. This clearly shows that standard linear consensus protocols are not resilient against malicious attacks.

B. W-MSR Guarantees Resilience Against 1 Malicious Robot

Consider the network as shown in Figure 4 with spoofed agent not being present. When the W-MSR resilient consensus algorithm (13) is used to obtain a robot formation, the legitimate robots are resilient against the malicious robot and achieve the desired formation in the safe region as shown in Figure 5a. This shows that the W-MSR is resilient against malicious attacks, but without spoofing.

C. The W-MSR Algorithm Fails Under a Spoofing Attack

Consider a spoofing attack on the network shown in Figure 5b, where agent 7 is malicious. When the malicious agent spoofs a single additional agent identity, the legitimate agents fail to achieve resiliency and hence they are pulled into unsafe region by malicious robots as shown in the Figure 5b. This shows that spoofing attacks are capable of compromising graph robustness properties and thereby the network resiliency.

D. Spoof Resilient W-MSR Achieves Desired Formation and Remains Safe

Consider a spoofing attack on the network shown in Figure 5c, where agent 7 is malicious spoofing agent 8. The spoofing attack is simulated for first 10 time steps and then the algorithm switches to spoof resilient version guaranteeing spoof resilient formation of legitimate robots in the safe region. The spoof detection needs to happen as early as possible or else the attack is capable of pulling the robot formation to unsafe region. Thus, under the proposed Spoof Resilient extension of the W-MSR algorithm, the legitimate agents achieve resilient formation, as shown in Figure 6c.
(a) Robots given random initial position in the safe region.

(b) Failure of linear consensus protocol with one malicious robot.

(c) W-MSR succeeds in keeping the formation in the safe region.

Fig. 5: With the robots given random initial positions as in (a), linear consensus protocol is not resilient against malicious attack as shown in (b). The W-MSR protocol guarantees a resilient formation against malicious robot in safe region as shown in (c). Blue and red colors represent legitimate and malicious robots, respectively.

(a) Robots are given random initial position in the safe region.

(b) The W-MSR algorithm fails under the spoofing attack.

(c) Our proposed modified W-MSR algorithm achieves spoof resiliency.

Fig. 6: With the robots given random initial positions as in (a), malicious robot (7) spoofs non-existent robot (8), compromising graph robustness properties. This resulted in the robot formation being pulled into unsafe region as shown in (b). As soon as the attack is detected, the malicious and spoofed robots are removed from the network and spoof resilient W-MSR guarantees resiliency against spoofing attack trying to pull the robot formation to an unsafe region as shown in (c). Blue, Red and Green colors represent legitimate, malicious and spoofed robots respectively.

E. Variations: Delayed and Probabilistic Spoof Detection

In practical settings, it may take a non-trivial amount of time to detect a spoofing attack, and spoofing may not be detected perfectly due to noise and uncertainty in the received signal properties. It has been shown in our previous work (23) that with longer delays in spoof detection, the malicious agents were able to cause larger deviations from the network’s final consensus value. Further, it was also shown that a malicious agent’s location in the networks affects how influential it can be in perturbing the state of the network; a spoof attack by agent 5 in Figure 4 with lower degree than agent 7 has a smaller transient impact. However, it is also possible to repair the impact of a spoofing attack on the network after delayed detection by maintaining memory of neighboring state transmission histories and subtracting out modifications made by spoofed and spoofing agents. Of course, such a bookkeeping effort would be limited by memory constraints, and may be cumbersome to implement in complicated networks. We are pursuing how this might be achieved in general for future work.

VI. EXPERIMENTAL RESULTS

To demonstrate the performance of the spoof resilient W-MSR algorithm, experiments are performed using a robotic platform to show that a team of robots can achieve a desired formation and remain in a safe region under the spoofing attack. In the experiments, each robot represents a node in the network, and the (2, 2)-robust network with six legitimate and one malicious agent, shown in Figure 4, is adopted to represent the communication among the robots. The objective of legitimate robots is to achieve a hexagon formation in a safe region, while the malicious agent aims to bring the formation to an unsafe region via the spoofing attack.

A. Implementation Details

Our experimental setup consists of seven Sphero 2.0 robots, a Logitech C950 webcam, a Bluetooth enabled smartphone, and a computer system which was equipped with Intel Core i7-6600U (4 CPU) @ 2.60 GHz processor. All routines executed during the experiment are implemented in MATLAB. We emphasize that although this experimental implementation is centralized, the information that is made available to each robot is restricted according to the communication graph shown in Figure 4. The webcam is set up to overlook a confined area in which the Sphero robots are placed and allowed to move. The LEDs of six legitimate robots are set to emit a blue color, while the malicious robot emits a red color. A color based image segmentation routine as explained in (32) is used to detect and track the robots in real-time from the 640 x 480 images that are fetched from the webcam. The idea behind the image reconstruction from webcam is shown in Figure 7a. The recovered coordinates from the camera data
are used in a consensus-based formation control strategy to bring the robots to a hexagon formation. As in the numerical simulations, at every iteration the desired position of robot \( i = 1, \ldots, 6 \) is computed via

\[
x_i[t + 1] = \sum_{j \in S_i[t] \setminus R[t]} w_{ij}[t] (x_j[t] - \bar{x}_j) + \bar{x}_i. \tag{17}
\]

where \( \bar{x}_i := [\sin(\theta_i), \cos(\theta_i)]^T \in \mathbb{R}^2 \), \( \theta_i := \frac{2\pi(i-1)}{6} \), is a constant bias vector that is used to position each robot at a vertex of the hexagon. For the demonstration purposes, a constant bias of \( \bar{x} = 50 \) cm is added to malicious robot’s position in each dimension to obtain spoofed robot’s position. Given the desired position \( x_i[t + 1] \), a low-level PID controller computes the required linear and angular velocity control commands that guide the robot to this locations at the next iteration. These control commands are communicated to the robots via Bluetooth as explained in (33). The motion of the legitimate robots are controlled by the computer and the malicious robot that aims to move the formation to unsafe region is controlled using smart phone. For all the experimental demonstrations, the robot swarms were started from the same positions as shown in the Figure 7b.

B. Observations

Using the 7-node \((2,2)\) robust network as shown in Figure 4, the W-MSR algorithm is implemented with seven robots, one of which is malicious and spoofed robot being absent. The algorithm as shown in Figure 8a is successful at ignoring the malicious robot’s intention to move the formation to an unsafe region and rest of the legitimate robots achieve the resilient formation. But when the W-MSR algorithm ran with the malicious robot spoofing one more additional identity, the assumption on the upper bound on the number of malicious agents in the network is compromised. As shown in the Figure 8b, the legitimate robots achieve the desired formation relative to the malicious robot and when the latter moves from the safe to the unsafe region, it is successful in pulling the network. In the Figure 8c, the legitimate robots perform fingerprint comparison test and detect the spoofing attack as early as possible and remove the malicious robot and its spoofed entities from the network. As a result, the legitimate robots achieve the formation in the safe region and remain resilient there unaffected by the spoofing attack.

VII. CONCLUSIONS

We proposed a spoof resilient consensus algorithm that extends a class of resilient consensus strategies, known as the Weighted Mean-Subsequence-Reduced (W-MSR) consensus, to provide resilience to malicious agents that may both adversely update state values and spoof non-existent agent identities. Physical fingerprint comparisons of received signals are used by legitimate agents to identify and isolate malicious agents that attempt spoofing attacks. The proposed algorithm using physical fingerprint approach guarantees resiliency despite the presence of a certain number of malicious agents and an arbitrary number of spoofed agents in the network. A probabilistic spoof detection analysis is presented using a semidefinite programming technique to arrive at distributionally robust Chebyshev bounds for probability of misclassification of robots. Experimental results using Sphero robot swarms and numerical simulations demonstrate the effectiveness of the proposed algorithm. The framework is applicable to a variety of problems involving multi-robot systems coordinating via wireless communication, including coverage, distributed estimation, and formation control. Future research involves investigating the spoof resiliency with different fault models and quantifying the worst-case probability of misclassifications persisting over extended periods of time by considering fingerprints over multiple time periods.
(a) Resiliency against malicious robot using WMSR algorithm.

(b) Spoofing attack results in failure of W-MSR resiliency.

(c) Spoof resiliency guaranteed with modified WMSR algorithm.

Fig. 8: A team of Sphero robots connected via Bluetooth as shown in Figure 4 is used to test the following scenarios. W-MSR algorithm is tested with a malicious attack as shown in (a). Then, the Spoofing attack is presented in (b) and finally the spoof resiliency is demonstrated in (c).

REFERENCES


