Distributed Direction of Arrival Estimation-aided Cyberattack Detection in Networked Multi-Robot Systems

Sangjun Lee and Byung-Cheol Min

***Abstract*— This study proposes a Direction of Arrival (DoA)-aided attack detection scheme to identify cyberattacks on networked multi-robot systems. For each agent, a local estimator is designed to generate robust residuals, and a parametric statistical tool corresponding to the residuals is elaborated to build sensitive decision rules. These locally stored residuals and thresholds are shared between robots via a wireless network, allowing a multi-robot system to complete its mission in the presence of one or more compromised agents. The proposed DoA-aided attack detection scheme is tested on a multi-robot testbed with a team of 10 robots. Experimental results demonstrate that the proposed detection scheme enables each robot to identify malicious activities without shearing the global coordination.**

I. Introduction

Cybersecurity is of fundamental importance to public safety and national security, as well as enabling innovation in cyberspace. While most robotic applications are increasingly dependent upon cyberspace, cybersecurity has not kept pace with the increase in cyber threats. A typical robotic application receives and transmits a great deal of information between sensors, actuators, controllers, and networks via cyberspace, all providing points of access for attackers. For this reason, units that govern safety should be protected from malicious activities, unauthorized access, and dubious activities, all of which could result in harmful outcomes. For example, an autonomous system's navigation system must be secured because it controls real-time position data directly linked to the physical behavior of the robot. A real-world example of a car hacking incident [1] showed the risks of not addressing these issues in current systems, and other examples [2], [3] in which unmanned aerial vehicles were captured and controlled via Global Positioning System (GPS) signal spoofing provide further proof of vulnerabilities and their consequences. These studies presented different cybersecurity issues and analyzed vulnerabilities that could result in worst-case scenarios.

A cyberattack, or simply an attack, is an action which undermines the security of robotic systems for malicious purposes. One approach to to guarantee security is a model-based one using fault detection, isolation, and reconfiguration methods [4], [5]. This is because a cyberattack can be treated as a random fault, which is additive or multiplicative. However, multi-robot systems suffer from specific vulnerabilities that classical control schemes are unable to fully address, necessitating the development of appropriate detection and identification techniques. For example, in the case of multi-robot systems, dependency on a central agent that transmits measurements and control signals increases the risk of cyberattacks [6], [7] if any single communication link is compromised, which would result in global task failure. To overcome this issue, each agent is required to have the capabilities of computing its own control input based on local information from on-board sensors or communicated by neighbors. Recent work in [8] presents a method that enables resilient formation control for mobile robot teams in the presence of defective robots. A different approach is used in [9], where a local adaptive fault observer and a distributed fault detection strategy were used to allow each robot to detect faults in other teammates even when not directly connected.

The use of an antenna-based approach for GPS spoofing detection and mitigation was presented in [10], where detection was based on comparing observed Directions of Arrival (DoAs) of satellite signals against predicted ones. A similar technique was implemented in [11] with an antenna array, and results practically demonstrated that observed DoAs can be used to identify spoofing attacks. On the other hand, information security methods from computer science, such as authentication, integrity, and cryptography techniques, appear inadequate for ensuring the security of robotic systems [12]. These methods are implemented without considering the underlying physical processes and control mechanisms relevant to robots.

In [13], we developed a path planning strategy that allowed multiple robots to reach their desired positions using DoA estimation. This work built upon the bearing estimation algorithm introduced in [14]. In another paper [15], an attack-aware multi-sensor integration scheme was presented for the detection of cyberattacks in autonomous vehicle navigation systems.

In this paper, we propose a DoA-aided attack detection method for networked multi-robot systems, with a focus on possible attacks on the navigation system. The navigation system is interrelated with the guidance system, which generates a path to the desired final destination. These systems should remain fully functional to ensure each robot's self-reliance and autonomy for the success of a mission. Thus, this study will determine if a robot's navigation system is being attacked. Any abrupt change or unexpected dynamic behavior will be identified by a local detection system. We assume that system alterations are caused by false data

This work was sponsored by fellowship 2017-R2-CX-0001 from the U.S. National Institute of Justice.

The authors are with the SMART Lab, Department of Computer and Information Technology, Purdue University, West Lafayette, IN 47907, USA lee1424@purdue.edu | minb@purdue.edu

injection attacks, corrupted signal readings, sensor failure, or any combination of these. The main contributions of this paper are summarized as follows:

1) Development of a distributed DoA-aided attack detection scheme for a network of multi-robot systems;
2) Generation of robust residuals with respect to DoA estimation in the presence of uncertainties;
3) Design of a parametric statistical test that enables the proposed system to quickly generate a detection alarm with low false alarm rate;
4) Verification of the proposed detection system in a multi-robot testbed.

The remainder of this paper presents the core components of the proposed DoA-aided detection scheme, organized in four sections. In Section II, multi-robot systems under attack are modeled as linear time-invariant systems subject to unknown attacks. In Section III, a DoA-aided attack detection scheme is developed using residual generation and threshold determination strategies. In Section IV, the proposed attack detection system is applied to a networked multi-robot system and experimentally validated. Lastly, conclusions and future research directions are discussed in Section V.

II. Problem Formulation

In this section, the mathematical models to describe the dynamics of a robot in networked multi-robot systems are presented, and a DoA estimation is developed to achieve attack detection.

A. Multi-Robot Systems

The system model that we consider is illustrated in Fig. 1. Within a robot, the detection scheme is initiated by receiving a control input and sensors then measure a sample of states. These states are fed into the state estimator to generate predictions. Lastly, the detector determines if there is an attack on the sensor through comparison between state estimations and actual measurements. At this time, residual information are shared across all the robots and each robot is able to selectively receive information from other robots.

A robot of multi-robot systems is described as a discrete liner time-invariant (LTI) system represented by a state-space model. The state-space model with given matrices A, B, and C is given as

$$x(k+1) = Ax(k) + Bu(k) + \nu(k) \quad (1)$$
$$y(k) = Cx(k) + \omega(k), \quad (2)$$

where $x \in \mathcal{R}^n$, $y \in \mathcal{R}^m$, and $u \in \mathcal{R}^r$ represent state vector, output vector, and control input vector, respectively, and where ν and ω are process and measurement noise that are represented by two independent Gaussian noise sequences with covariance matrices Q and R, respectively. If a sensor of the i-th robot for $i = \{1, 2, 3, ..., N\}$ given N number of multiple robots is being compromised which means that unknown signals have been injected, added, or modified to

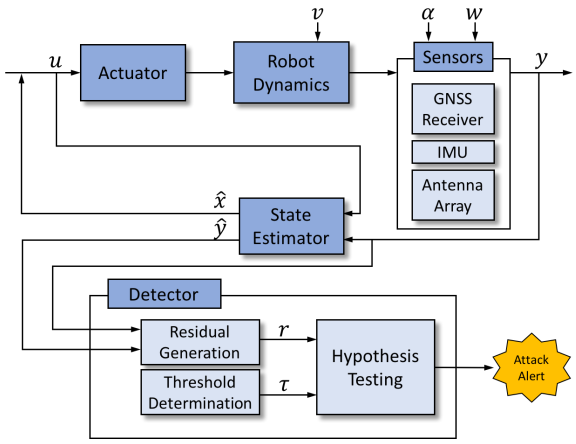


Fig. 1. Schematic overview of the detection system that each robot runs in the networked multi-robot system.

the sensor, the LTI system (1) and (2) can be written as follows:

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k) + \nu_i(k)$$
$$y_i^{\alpha_i}(k) = C_i x_i(k) + \alpha_i(k) + \omega_i(k), \quad (3)$$

where $\alpha_i \in \mathcal{R}^m$ denotes additive attacks on a sensor and the state with the superscript α_i represents the system after an attack occurs. The key idea behind this is that the differences induced by attacks would be detectable from the proposed scheme in the presence of uncertainties.

B. Attack Model

False data injection attacks defined in [16] are output attacks that render an unstable mode (if any) of the unobservable system. False data injection attacks refer to attacks that compromise the integrity of control packets or measurements, and they are cast by altering the behavior of sensors and actuators. The measurement model of a sensor for false data injection attack becomes:

$$z_i^{\alpha_i}(k) = C_i x_i(k) + \alpha_i(k) + \omega_i(k) \quad \text{for } k = k_\alpha,$$

where α_i is the malicious offset injected by the attacker at k_α.

C. Measurement Models

Two typical navigation solutions of mobile robots, Inertial Navigation System (INS) and Global Navigation Satellite System (GNSS) measurements, plus antenna array measurements are considered. An INS uses an Inertial Measurement Unit (IMU) to track the position, velocity, and orientation of a vehicle relative to an initial point, orientation, and velocity. A GNSS provides satellite signals that can be processed in a GNSS receiver, allowing the receiver to estimate its current position and velocity. In addition, an antenna array provides the direction of arrival of received signal which allows a robot to estimate the bearings of neighboring robots. There are no states directly affected by the INS measurements, the GNSS measurements, or the DoA measurements in the

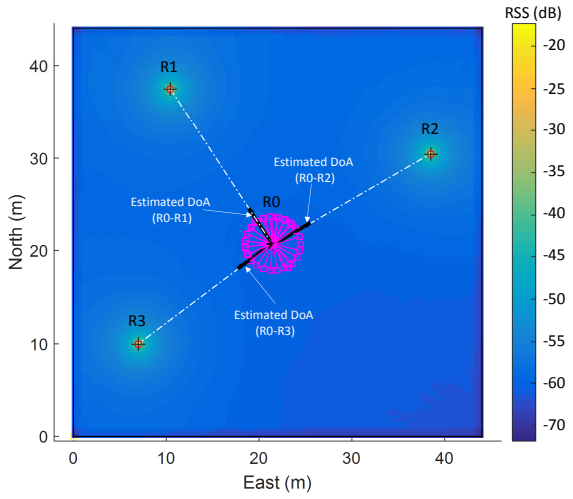


Fig. 2. An illustrative example of the DoA estimations for networked multi-robot systems. R0 estimates the other three robots (R1, R2, and R3) using the weighted centroid algorithm proposed in [14]. The white dotted lines indicate actual DoAs and the black arrows indicate estimated DoAs. The color bar on the right side represents the level of received signal strength. Bright yellow represents stronger signal strength than blue.

system equation (1), but they interact through the output equation (2) determined by the measurement models:

$$z_i = \begin{bmatrix} z_{\text{GNSS},i} \\ z_{\text{INS},i} \\ z_{\text{DoA},i} \end{bmatrix}.$$

A probabilistic model-based approach is proposed to estimate DoA to radio signal sources, which are generated by other robots in a networked multi-robot systems. This approach allows a robot to mitigate estimation errors that are significantly affected by its environments. Thus, a robot takes Received Signal Strength (RSS) measurements via an array of directional antennas and then finds the best angle that is a mean of weighted centroid approaches. A DoA of the i-th robot for the given number of measurements M is determined by:

$$\hat{\theta}_i(k) = \sum_{j=1}^{M} \beta_i(j)\theta_i(j) \Big[\sum_{j=1}^{M} \beta_i(j) \Big]^{-1}, \tag{4}$$

where β_i is a weight computed by the RSSs at the j-the measurement, that is the solution of $log_{10}\beta_i = RSS_i(j)/\gamma_i$ with a positive gain γ. This enables a stronger signal strength with more weight than a weak signal strength. An illustrative example is shown Fig. 2. In this figure, a robot receives the DoA information from three other sources and estimates their local location using (4). The DoA estimation will be used to design an attack detection scheme in the following section.

III. Attack Detection and Identification

In this section, a stochastic system is considered to model a multi-robot system for the solutions of the attack detection and identification problems. The task is comprised of two subsections: Kalman filter-based residual generation and a decision rule using statistical change detection algorithms.

A. Residual Generation

An innovation filter is designed to determine a stable linear time-invariant filter with output signals such that:

Condition 1. If there is no attack ($\alpha_i(k) = 0$ for all k), output vector y will be a zero mean white noise vector that is not affected by u;

Condition 2. If there is any attack ($\alpha_i(k) \neq 0$ for $k \geq k_\alpha$), output vector will not be a zero mean white noise vector that is affected by the unknown input vector u.

Under the assumption that the system will stay in the steady-state until attacked, a steady state Kalman filter can be used. This is because an innovation filter provides a output prediction $y_i(k)$ and $\hat{y}_i(k)$. It enables the system to identify any abrupt changes on sensor measurements. Estimator dynamics provided by the following steady-state Kalman filter is considered:

$$\hat{x}_i(k+1) = A_i\hat{x}_i(k) + B_iu_i(k) + L_i[y_i(k) - \hat{y}_i(k)],$$

where Kalman gain is $L_i = P_iC_i^T(C_iP_iC_i^T + R_i)^{-1}$ with the covariance matrix given by $P_i = A_i[P_i - P_iC_i^T(C_iP_iC_i^T + R_i)^{-1}C_iP_i]A_i^T + Q_i$. Note that the detectability of (A_i, C_i) ensures the existence of such an estimator. This integration provides continuous position estimation, and the controller receives it to achieve the desired path. The following output feedback controller is considered:

$$u_i(k) = u_{\text{ref},i} + K_i[\hat{x}_i(k) - x_{\text{ref},i}(k)],$$

where $x_{\text{ref},i}$ is the reference state corresponding to the reference input $u_{\text{ref},i}$ and K_i represents the feedback gain matrix.

The main purpose of the detection capability is to generate robust residuals to uncertainties and determine sensitive thresholds to false alarm. As shown in Fig. 1, the detector determines the system condition at each time step through statistical hypothesis testing that compares the residual and threshold generated. The residual is the difference between the actual measurements and the estimates. A sequence of the residuals for the i-th robot is defined as

$$r_i(k) = y_{\alpha,i}(k) - \hat{y}_i(k). \tag{5}$$

The residuals evolve with the output estimate given by $\hat{y}_i(k) = C_i\hat{x}_i$ and the estimation error defined as $e_i(k) = x_i - \hat{x}_i$. Then, the output prediction error, which is the innovation to the standard Kalman filter, is defined as:

$$r_i(k+1) = C_ie_i(k+1) + \alpha_i(k+1),$$

where the estimation error dynamics are given by $e_i(k+1) = (A_i - L_iC_i)e_i(k)$. It can be used to obtain the new information in $y_i(k)$, which was not available in $y_i(1), \ldots, y_i(k-1)$. Therefore, *Condition 1* is satisfied when the innovation vector is a white noise vector that is not affected by u. Similarly, *Condition 2* is also satisfied by the innovation sequence when the attack information is identified in the innovation vector.

B. Decision Rule

In our system where the system is modeled as a linear stochastic model, an attack detection problem can be solved by sequential change detection algorithm with appropriate hypotheses. Consider a series of independent and distributed random vectors $z(k) \sim \mathcal{N}(\mu, Q)$. If there is no attack before an unknown attack time, μ is equal to μ_0. On the other hand, it would change to $\mu = \mu_1 \neq \mu_0$ if there is an attack at time k_α. The change detection problem is to identify the difference between when the system is normal ($\mu = \mu_0$) and the parameter μ has changed to μ_1 due to an attack. Thus, the detection problem is to distinguish between two hypotheses: $\mathcal{H}_0$–the normal case, $\mathcal{H}_1$–the abnormal case where a change has taken place. For the condition under $\mathcal{H}_0$, the parameter $\boldsymbol{\mu}_0$ is computed based on the system model under normal operation. The cumulative sum (CUSUM) algorithm [17] is used to detect a known change regardless of the availability of prior knowledge about the system probability. Build the following two-sided hypotheses test with $\mu_0 \neq \mu_1$:

$$\begin{aligned} \mathcal{H}_0 :&\quad z(k) \sim \mathcal{N}(\mu_0, Q) \text{ for } k = 1, ..., k \\ \mathcal{H}_1 :&\begin{cases} z(k) \sim \mathcal{N}(\mu_0, Q) \text{ for } k = 1, ..., k_\alpha - 1 \\ z(k) \sim \mathcal{N}(\mu_1, Q) \text{ for } k \geq k_\alpha. \end{cases} \end{aligned}$$

In order to estimate the change time and its magnitude, the probability density function of a Gaussian vector z is defined as:

$$p_\mu(z) = \frac{1}{\sqrt{(2\pi)^n \det Q}} \exp\left(-\frac{1}{2}(z-\mu)^{\mathrm{T}} Q^{-1}(z-\mu)\right),$$

where μ is the mean and Q is the variance. The log-likelihood ratio can be represented as:

$$\begin{aligned} S(z(k)) &= \ln \frac{p_{\mu_1}(z(k))}{p_{\mu_0}(z(k))} \\ &= (\mu_1 - \mu_0)^{\mathrm{T}} Q^{-1}\left(z(k) - \frac{1}{2}(\mu_0 - \mu_1)\right). \end{aligned}$$

In case of the system described in (3), the recursive computation of the CUSUM decision function can be performed as:

$$S_i(k+1) = \begin{cases} \max\left(0, S_i(k) + |r_i(k+1)|\right) \text{ if } S_i(k) \leq \tau_i(k) \\ 0 \text{ and } k_\alpha = k \text{ if } S_i(k) > \tau_i(k). \end{cases} \tag{6}$$

The null hypothesis is rejected if the test statistics S_i is greater than the threshold τ_i. In this case, the test provides a global attack alarm time k_α that is the smallest time instance at which S_i exceeds a given threshold, and the test starts over. The null hypothesis is accepted if the test statistics S_i is less than or equal to the threshold τ_i. The test continues without stopping in this case. In practice, this test collects a number of samples and calculates their weighted sum to detect a significant change in the mean of samples.

IV. Experiment

This section presents experiments with the Robotarium testbed to evaluate the proposed DoA-aid attack detection system. The Robotarium [18], [19] is a multi-robot testbed developed at the Georgia Institute of Technology.

A. Implementation

An experiment is designed to test if the proposed detection system identifies attacks on multiple robots. Each of the 10 robots was spawned at a random pose, and tried to complete a global goal of reaching one common destination. A separate function from the detection system injected the attacks into the pose measurements of 3 arbitrarily selected robots when the global clock reached 7 seconds. A collision avoidance was executed by default and the equation of motion for the i-th robot was governed by the following dynamics:

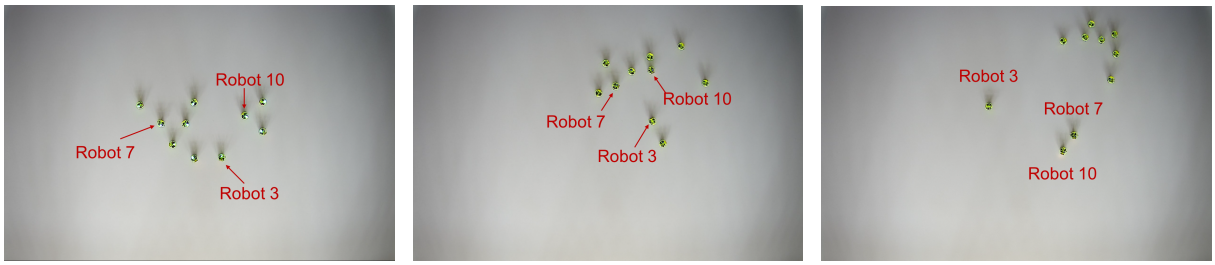
$$\begin{aligned} \dot{x}_i &= v_i \cos\theta_i \\ \dot{y}_i &= v_i \sin\theta_i \\ \dot{\theta}_i &= w_i, \end{aligned}$$

where x_i and y_i represent the position of the i-th robot along its local eastern axis and its local northern axis, respectively, and θ_i, v_i, and w_i are the orientation of the robot, its linear velocity, and angular velocity, respectively.

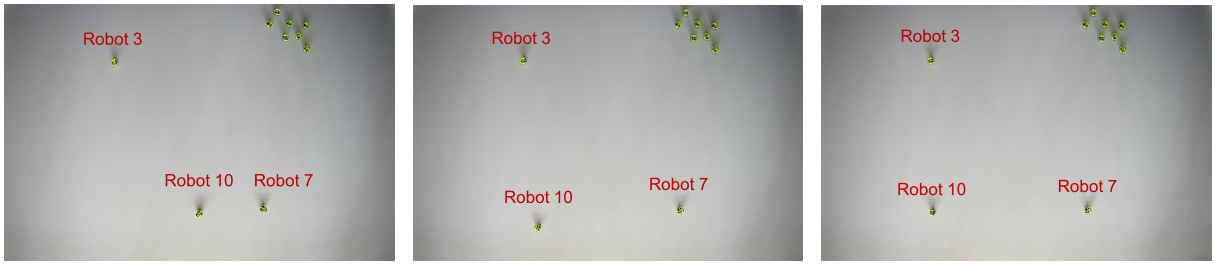
For the experiment, the continuous time state equations were discretized with the sampling time T, which produced the nonlinear discrete-time state model and the linear measurement model under normal operation. These models were linearized to correspond with the state-space model in (1) and (2) by using the state and measurement Jacobian matrices. In addition, initial states $x_i(0)$, state error covariance P_i, process noise covariance Q_i, and measurement noise covariance R_i were carefully chosen according to hardware specifications. An extended Kalman filter was performed to predict the robot states under normal operation. This integrated architecture ensured that a continuous navigation solution was always produced, regardless of the existence of attacks. Following the state estimation under normal condition, the system under attack (3) was considered. These two different measurement models were used for the residual generation in (5). The decision rules in (6) then determined if there was a significant change in the robot position at each time step.

B. Results

The chronological sequence of the robots' configurations during the experiment is illustrated in Fig. 3. Each of the 10 robots set the final destination to $(0.9, 0.9)$ as a common goal. Then, they were deployed from a random pose. Actual trajectories of the robots are shown in Fig. 4. In this figure, three of the robots were clearly set apart from the team and failed to achieve the common task due to the attacks while others succeeded. This was mainly because each robot computed control commands completely based on its local information so that each robot was unable to verify if the system was functioning properly unless it shared



(a) Initial pose (b) 9 seconds (c) 18 seconds



(d) 27 seconds (e) 36 seconds (f) Final pose

Fig. 3. Robot configurations by time. (a) Each of the 10 robots was initially deployed at a random pose. (b) A random amount of attacks with boundary limits were injected to 3 random robots at 7 seconds. (c) Robots 3, 7, and 10 were heading to new destinations. (d) The compromised robots were separated from the team. (e) The compromised robots were unable to verify if the current control input was proper unless each robot sheared the global position of the entire team. (f) The compromised robots stopped at incorrect destinations while the rest of the team achieved the common task.

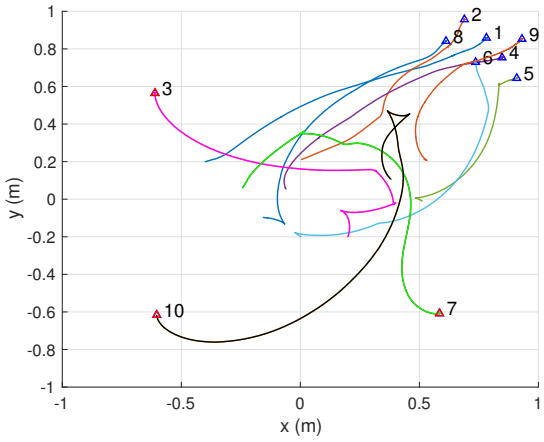


Fig. 4. Actual trajectories of the robots during the experiment. Each robot started from its initial location and followed its desired path to reach a common point. The final location of each robot is illustrated using triangles. Injected attacks caused the failure of Robots 3, 7, and 10 in the red triangle while other robots represented in the blue triangle succeeded.

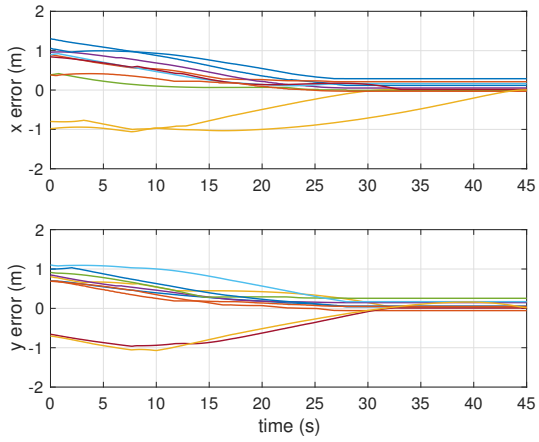
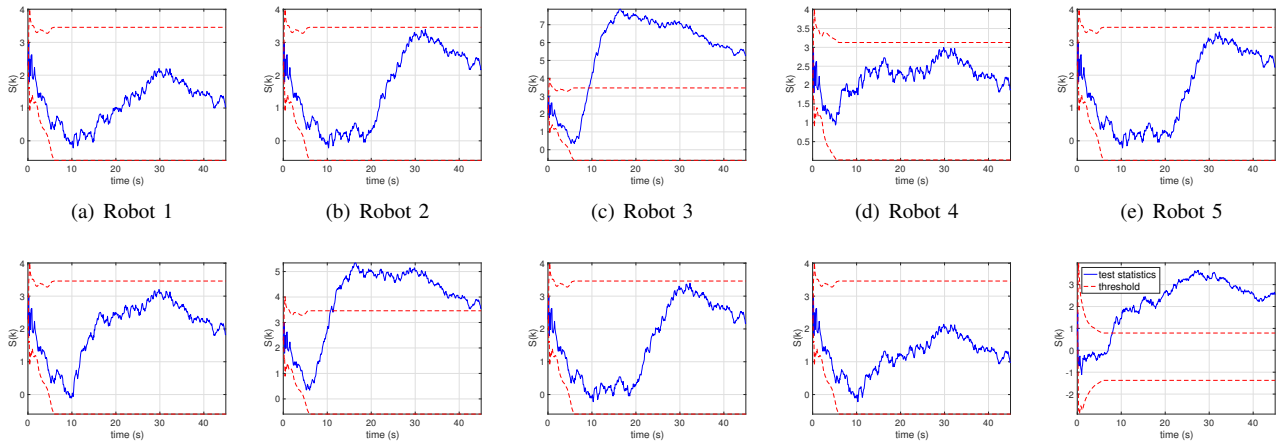


Fig. 5. Position tracking error of the robots. The control commands for each robot performed well but no evidence of malicious activities was indicated here. This was because the current control scheme was unable to identify any attack without the aid of the detection system.

the global coordination. For example, the position tracking error of the robots in Fig. 5 converges to zero even though attacks occurred. Thus, it was unable to identify the system's functionality based on this information.

However, the evolution of the test statistics in Fig. 6 clearly shows that there were significant changes to Robots 3, 7, and 10 that caused the residual to jump the upper bound of the threshold around the 10 second mark. The test statistics were calculated by (6), and the upper and lower bounds of the threshold were generated by using the weighted sum of the first 15 samples of sensor measurements. Based upon these parameters, the detector in each robot determined that there was an attack when the residual went above the upper limit of the threshold, and the corresponding time was automatically generated. The generated times were $t_{\alpha,3} = 9.9$ sec, $t_{\alpha,7} = 9.4$ sec, $t_{\alpha,10} = 8.5$ sec., showing relatively quick detection because they were only a few sampling steps behind the actual attack. In addition, there were a number of ups and downs for Robots $1, 2, 4, 5, 6, 8$, and 9 prior to the attack, but they stayed within the threshold boundary, allowing the detection algorithm to avoid a false alarm. Thus, based on the experiment, using the proposed DoA-aided attack detection



(a) Robot 1 (b) Robot 2 (c) Robot 3 (d) Robot 4 (e) Robot 5

(f) Robot 6 (g) Robot 7 (h) Robot 8 (i) Robot 9 (j) Robot 10

Fig. 6. Test statistic evolutions of the robots. In (c), (g), and (j), the proposed system identified a significant change of the residuals that exceeded the upper limit of the threshold. On the other hand, there was no attack alarm for the rest of the robots.

scheme provides a solution to detect attacks on multiple robots as quickly as possible.

V. Conclusions

This research has presented a direction for arrival-aided cyberattack detection on networked multi-robot systems. Starting with a state-space model of a system under attack, a parametric statistical tool to generate a decision rule was developed with an innovation filter. This approach provided real-time attack detection for a robot in a networked multi-robot system, protecting against possible cyberattacks. A team of robots with the presence of multiple compromised agents was employed on a multi-robot testbed to test the performance of the proposed scheme.

References

[1] C. Miller and C. Valasek, "Remote exploitation of an unaltered passenger vehicle," *Black Hat USA*, vol. 2015, 2015.

[2] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "Unmanned aircraft capture and control via gps spoofing," *Journal of Field Robotics*, vol. 31, no. 4, pp. 617–636, 2014.

[3] D. P. Shepard, T. E. Humphreys, and A. A. Fansler, "Evaluation of the vulnerability of phasor measurement units to gps spoofing attacks," *International Journal of Critical Infrastructure Protection*, vol. 5, no. 3, pp. 146–153, 2012.

[4] S. Z. Yong, M. Zhu, and E. Frazzoli, "Resilient state estimation against switching attacks on stochastic cyber-physical systems," in *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*. IEEE, 2015, pp. 5162–5169.

[5] A. Teixeira, S. Amin, H. Sandberg, K. H. Johansson, and S. S. Sastry, "Cyber security analysis of state estimators in electric power systems," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 5991–5998.

[6] M. Amoozadeh, A. Raghuramu, C.-N. Chuah, D. Ghosal, H. M. Zhang, J. Rowe, and K. Levitt, "Security vulnerabilities of connected vehicle streams and their impact on cooperative driving," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 126–132, 2015.

[7] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[8] K. Saulnier, D. Saldana, A. Prorok, G. J. Pappas, and V. Kumar, "Resilient flocking for mobile robot teams," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1039–1046, 2017.

[9] F. Arrichiello, A. Marino, and F. Pierri, "Distributed fault detection and recovery for networked robots," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 3734–3739.

[10] M. Appel, A. Konovaltsev, and M. Meurer, "Robust spoofing detection and mitigation based on direction of arrival estimation." in *Proceedings of the 28th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2015)*, 2015.

[11] M. Cuntz, A. Konovaltsev, M. Heckler, A. Hornbostel, L. Kurz, G. Kappen, and T. Noll, "Lessons learnt: The development of a robust multi-antenna gnss receiver," *Proc. ION GNSS 2010*, pp. 21–24, 2010.

[12] D. I. Urbina, J. Giraldo, A. A. Cardenas, J. Valente, M. Faisal, N. O. Tippenhauer, J. Ruths, R. Candell, and H. Sandberg, *Survey and new directions for physics-based attack detection in control systems*. US Department of Commerce, National Institute of Standards and Technology, 2016.

[13] B.-C. Min, R. Parasuraman, S. Lee, J.-W. Jung, and E. T. Matson, "A directional antenna based leader–follower relay system for end-to-end robot communications," *Robotics and Autonomous Systems*, vol. 101, pp. 57–73, 2018.

[14] B.-C. Min, E. T. Matson, and J.-W. Jung, "Active antenna tracking system with directional antennas for enhancing wireless communication capabilities of a networked robotic system," *Journal of Field Robotics*, vol. 33, no. 3, pp. 391–406, 2016.

[15] S. Lee, Y. Cho, and B.-C. Min, "Attack-aware multi-sensor integration algorithm for autonomous vehicle navigation systems," in *Systems, Man, and Cybernetics (SMC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 3739–3744.

[16] Y. Mo, E. Garone, A. Casavola, and B. Sinopoli, "False data injection attacks against state estimation in wireless sensor networks," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 5967–5972.

[17] E. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, no. 1/2, pp. 100–115, 1954.

[18] D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, and M. Egerstedt, "The robotarium: A remotely accessible swarm robotics research testbed," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1699–1706.

[19] D. Pickem, M. Lee, and M. Egerstedt, "The gritsbot in its natural habitat-a multi-robot testbed," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4062–4067.