Cumulative Sign Detector for Stealthy Attacks on Mobile Robotic Swarms

Paul J Bonczek[1] and Nicola Bezzo[1,2,3]

[1]Department of Electrical and Computer Engineering, [2]Department of Systems and Information Engineering, [3]Department of Computer Science
University of Virginia

MOTIVATION

- Mobile robotic swarms are susceptible to stealthy attacks (such as system hijacking) that can affect its swarming behavior and performance.
- Smart attackers are able to learn and leverage information about the robot system models and noise characteristics to develop stealthy attacks.
- Attacks intending to hijack a mobile robot will leave traces of non-random behavior that contradict model and swarm behaviors.

OBJECTIVE

- Monitor information for **non-random** and **inconsistent** behavior.
- **Detect** and **isolate** compromised vehicles from the robotic swarm.
- **Maintain** a task, such as performing go-to-goals operations.

VIRTUAL SPRING-MASS SYSTEM

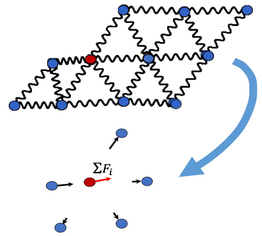
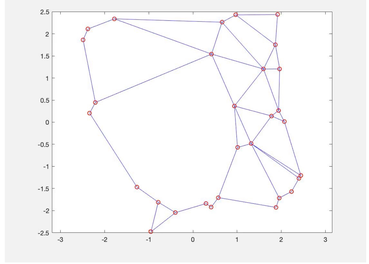


Figure 1. Swarming formation follows a network of virtual spring-masses.

$$\ddot{X}_i = \left[\sum_{j \in \mathcal{S}_i} k_{ij}\left(l_{ij} - l_{ij}^0\right)\hat{\mathbf{d}}_{ij}\right] - \gamma_i \dot{X}_i$$

Equation 1. Acceleration is determined by the sum of all spring forces.



Video 1. Virtual Spring-Mass example.

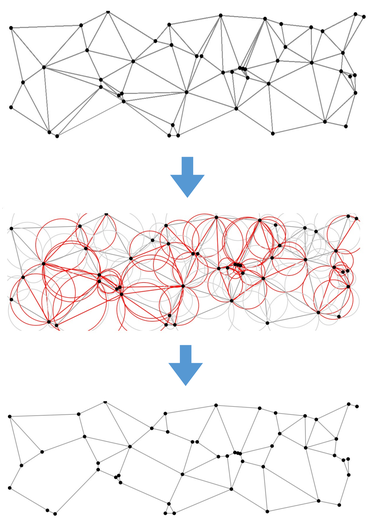


Figure 2. (top) Delaunay Triangulation creates edges [*springs*] between nodes [*circles*]. (middle) Draw circumcircles along Delaunay edges, eliminate edges if other nodes reside inside [red] circumcircle. (bottom) After eliminating non-compliment edges, resulting node/edge network is the final Gabriel Graph for virtual spring-mass network.

CONSIDERED CONSTRAINTS AND ATTACKS

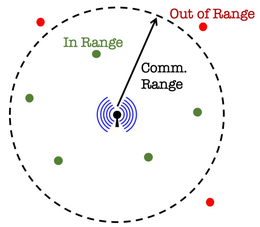
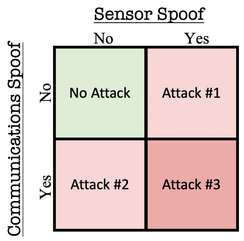


Figure 3. (left) limited communication range to neighboring vehicles.

Figure 4. (right) Three types of feasible attacks are considered, to on-board sensors and/or spoofed communication broadcasts.



Information Sharing: Each vehicle broadcasts on-board measurements and position, along with its neighbor's positions and nearby obstacles and goal points used for control.

Listening Nodes: Each vehicle "listens" to neighboring vehicles determined by edges of Gabriel Graph. Received information is used in position prediction of neighbors.

CONTROL AND DETECTION ARCHITECTURE

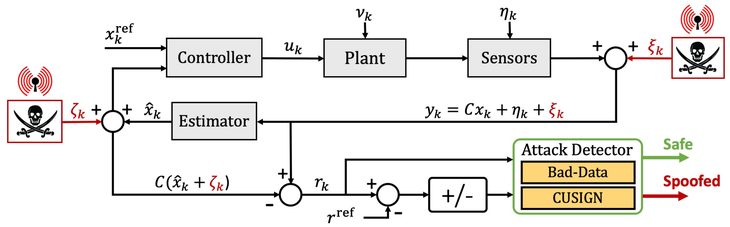


Figure 5. Control system architecture of a with potential false data entry points on measurements and state estimates. Residual-based detection scheme for detection.

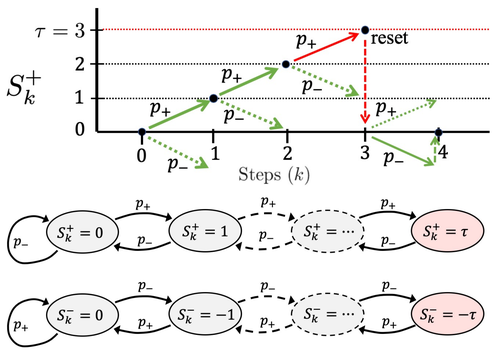


Figure 6. CUSIGN accumulates the sequence of signed residual values with respect to a reference point.

Figure 7. CUSIGN test variables can be described as Markov Chain, with the threshold terminal state (red) triggering an alarm.

SIMULATION RESULTS

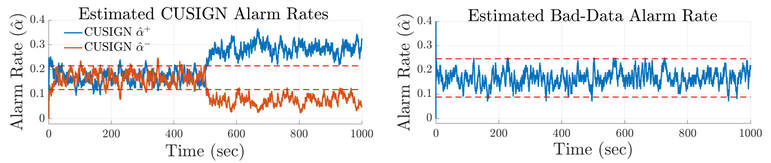
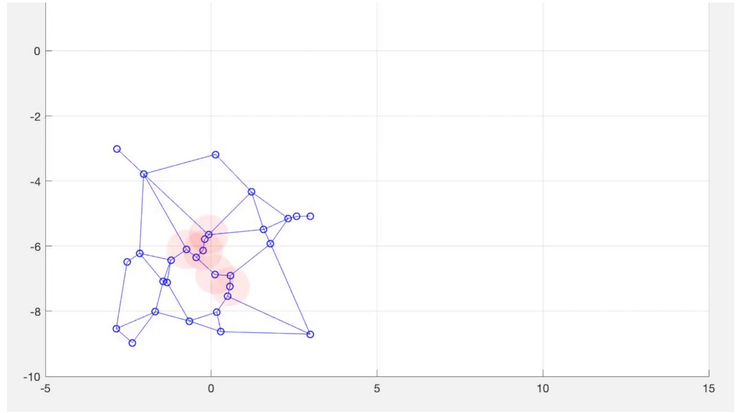


Figure 8. Results for bias injection onto a sensor measurement at 500 seconds. CUSIGN detects non-random behavior while Bad-Data does not, remaining within bounds.



Video 2. Swarm performing a go-to-goal task with two compromised vehicles, then a Leader-follower swarm performing go-to-goal tasks under the effect of attacks.

CURRENT AND FUTURE WORK

➢ Explore methods to predict vehicle positions of compromised "rogue" vehicles, which broadcast incorrect information to its neighbors. If neighboring vehicles can accurately predict rogue vehicle movements, swarming behavior can become safer and exhibit more ideal behavior.

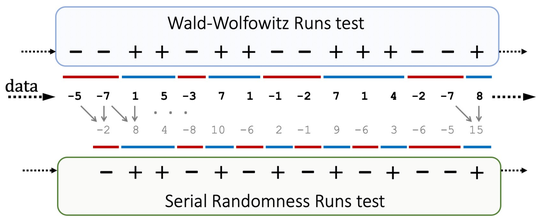


Figure 9. Develop a run-time monitor using Randomness of Runs tests without the need to store entire sequences of data for detection of hidden, malicious attacks.