Resilient Active Target Tracking with Multiple Robots

Lifeng Zhou[1], Vasileios Tzoumas[2], George J. Pappas[3], and Pratap Tokekar[1]
[1]Virginia Tech, USA, [2]MIT, USA, [3]UPenn, USA

This work was partially supported by the ARL DCIST CRA W911NF-17-2-0181 program, and by the NSF grants 1566247 and 1637915

Virginia Tech — Invent the Future®
Massachusetts Institute of Technology
Penn Engineering | GRASP Laboratory — General Robotics, Automation, Sensing & Perception Lab

MOTIVATION

- **Objective:** Plan the motion of multiple robots to actively track the targets.
- **Challenges:** Adversaries can attack robots, block, or compromise their tracking sensors (Fig. 1).

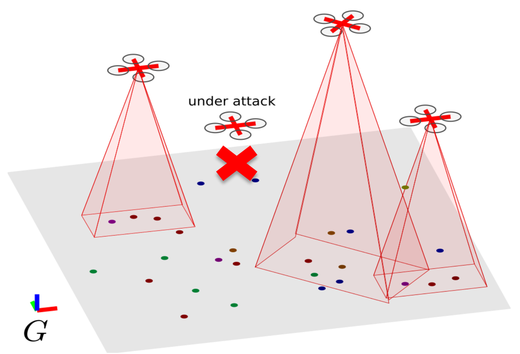

Fig. 1. Aerial robots mounted with down-facing cameras so to track multiple targets on the ground. If a robot is under attack, its camera will be blocked.

PROBLEM FORMULATION

- **Problem (Resilient Multi-Robot Multi-Target Tracking):**

$$\max_{\mathcal{S}\subseteq\mathcal{X}} \ \min_{\mathcal{A}\subseteq\mathcal{S}} \ f(\mathcal{S}\setminus\mathcal{A})$$

$$s.t.\ \mathcal{S}\in\mathcal{I},\ |\mathcal{A}|\leq\alpha\leq N$$

- f : a target tracking objective that is monotone and submodular (e.g., the expected number of covered targets in current round).
- $\mathcal{X}$: a ground round set of all trajectories from N robots.
- $\mathcal{I}$: a natural constraint that each robot follows one trajectory.
- $\mathcal{S}$: the set of selected trajectories for N robots.
- $\mathcal{A}$: a set of trajectories removed / attacked from the selected set.

EVALUATION

- **Simulation setting:** 4 aerial robots are tasked to track 30 ground targets (Fig. 3 (a) & (b)) . Each robot has 4 candidate trajectories (Fig.3 (c)). The robot will follow one at each time step.

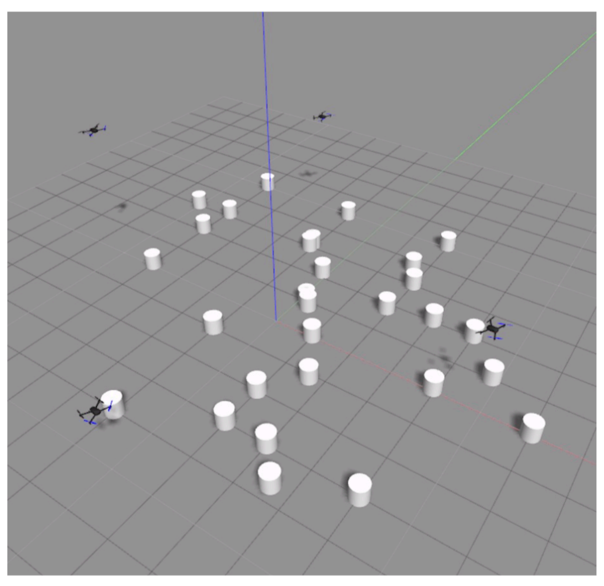

Fig. 3. (a) Gazebo environment.

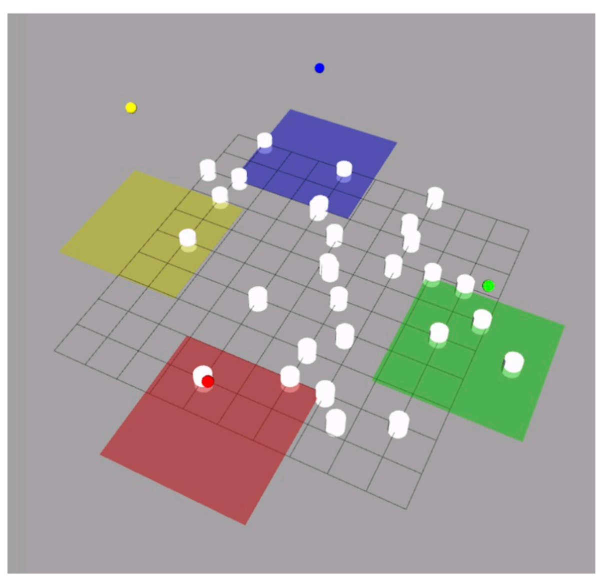

Fig. 3. (b) Rviz environment.

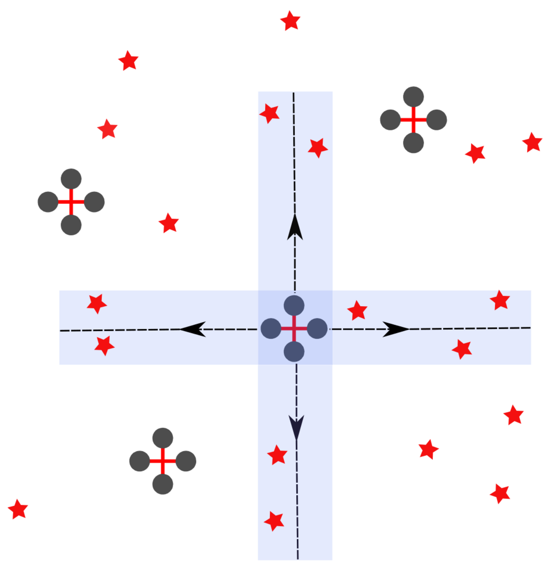

Fig. 3. (c) A top view of robots and targets.

- **Comparison of the number of targets covered after the attacks: (1) optimal, (2) resilient, (3), greedy and (4) random.**

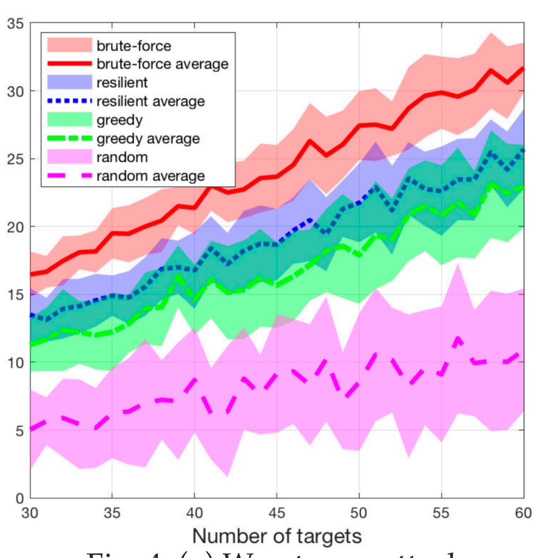

Fig. 4. (a) Worst-case attack.

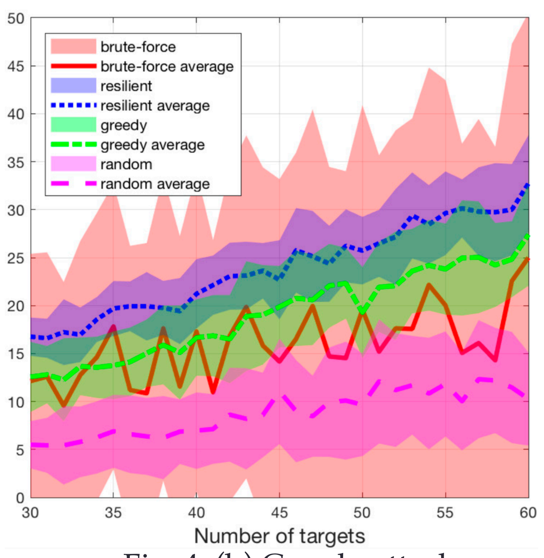

Fig. 4. (b) Greedy attack.

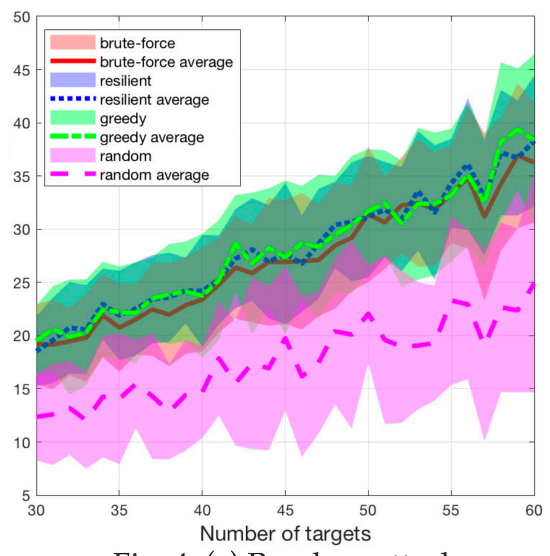

Fig. 4. (c) Random attack.

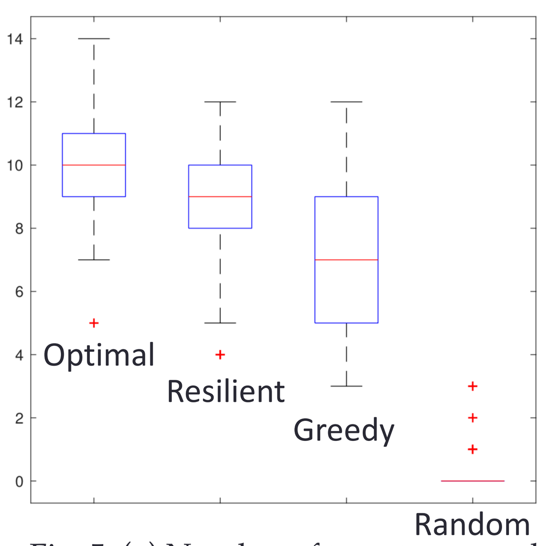

Fig. 5. (a) Number of targets covered.

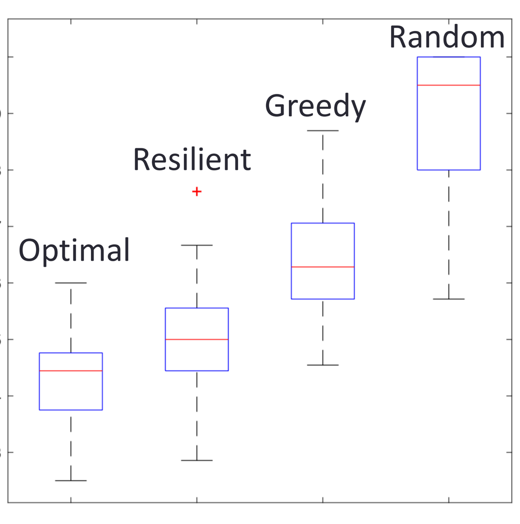

Fig. 5. (b) Attack rate.

- Resilient performs close-to-optimal, and better than greedy and random.
- Resilient is also robust to non-worst-case attacks and no-attacks than greedy and random.

FRAMEWORK

- **Robots/sensors:** Team of robots carrying onboard sensors (e.g., cameras) that can track targets.
- **Trajectories:** At each round, each robot generates a set of candidate trajectories. Among them, the robot will choose one (Fig. 2).
- **Targets:** Targets exists in an area of interest. The targets can be mobile or immobile.
- **Attacks:** At any time, at most α robots will fail due to attacks.

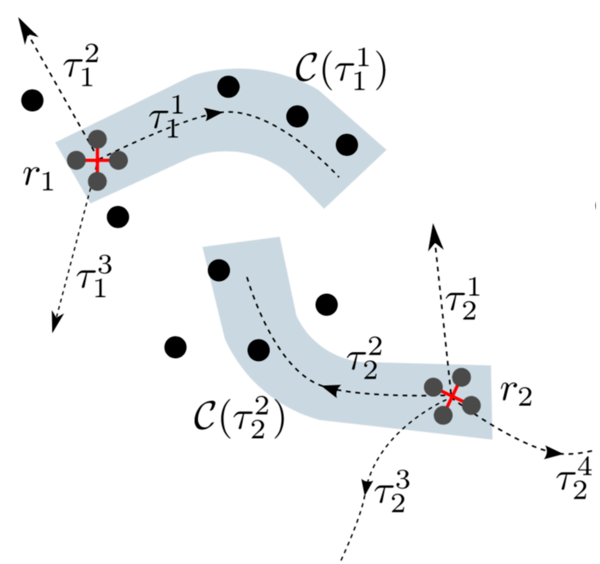

Fig. 2. Each robot has a set of candidate trajectories —dotted lines with arrow. Each trajectory covers a certain number of targets—black dots. The highlighted regions are the tracked regions of the associated trajectories.

CONTRIBUTIONS

- **Resilient target tracking algorithm:**
  - **Oblivious decision :** robots rank the α most profitable trajectories.
  - **Greedy decision:** remaining $N-\alpha$ robots choose trajectories greedily, assuming the robots chosen in the oblivious decision set will all be attacked.
- **Performance of resilient algorithm:**
  - **Approximation performance:** k_f is the curvature of f

$$\frac{f(\mathcal{S}\setminus\mathcal{A}^\star(\mathcal{S}))}{f^\star}\geq\frac{1}{1+k_f}\max\left[1-k_f,\frac{1}{1+\alpha},\frac{1}{N-\alpha}\right]$$

  - **Running time:** constructs set $\mathcal{S}$ in $O(N^2D^2)$ time. D is the number of candidate trajectories for each robot.
- **Analysis of the performance:**
  - Approximation ratio is always larger than 0 with finite number of robots, N.
  - When $k_f=0$, resilient algorithm gives optimal solution.
  - When no failures ($\alpha=0$), algorithm has same performance as the non-resilient greedy algorithm [Conforti & Cornuejols, DAM '84].
  - Resilient runs as fast as the greedy algorithm.

EXTENSIONS

- **Online resilient optimization:**
  - The current method is an offline planner that cannot react to real-time failures.
- **Distributed resilient optimization:**
  - Adversaries attack a number of robots.
  - Robots have a limited communication range.

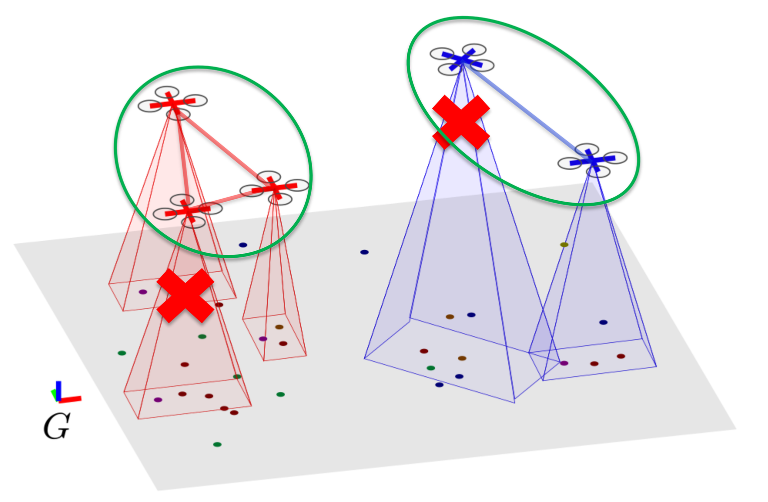

Fig. 6. Distributed Resilient.

L. Zhou and P. Tokekar. **An Approximation Algorithm for Distributed Resilient Submodular Maximization**
ICRA workshop 2019, Resilient Robot Teams: Composing, Acting, and Learning.
13:30-17:30 | Friday 24 May | Room 518b