(Byzantine) Attacks and Defenses in Multi-Robot Systems

Cristina Nita-Rotaru

Khoury College of Computer Science

Northeastern University

The robots are here

- Search and Rescue
- Precision Agriculture
- Warehouse Automation
- Environmental Monitoring
- Manufacturing and Assembly
- Exploration of Unknown Environments
- Surveillance and Security
- Medical Applications
- Traffic Management
- Collaborative Mapping
- Entertainment and Education
- Aerial Swarm Applications

Multi-robot systems

Multiple robots collaborate and communicate to achieve common goals

Centralized architecture

- A single entity makes decisions for all robots in the system
- Gathers information from each robot, processes it, and issues commands
- Simplifies coordination, but the central entity is a single point of failure

Distributed architecture

- Robots collaborate to make collective decisions
- Robots communicate to share information
- No single point of failure, but coordination is more complex

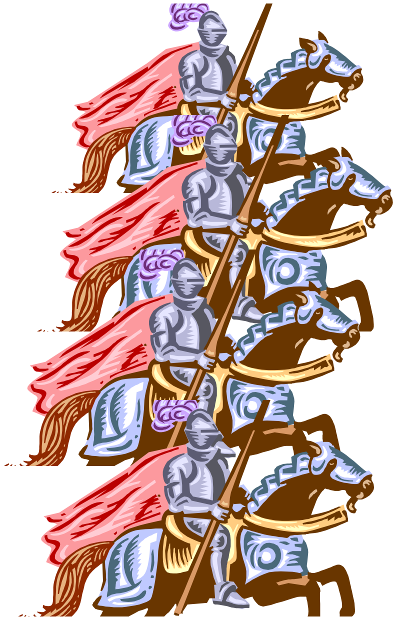
Vulnerable to cyber attacks

Multi-robot systems rely on communication networks and software systems, thus they are susceptible to cyber attacks

- Such attacks can result in:
  - Unauthorized access
  - Compromising confidentiality
  - Degradation of task reliability and accuracy
  - Destruction through unintended collision
  - Safety of surrounding environment
  - Safety for humans working in proximity

Byzantine adversaries

- Cooperative applications are susceptible to compromised on malfunctioning devices
    - Robots can get compromised or malfunction
- Byzantine adversaries models compromised or malfunctioning parties
    - Takes the form of lying, two-face behavior, dropping (not providing) information, colluding for stronger attack and avoiding detection



Challenges for dealing with Byzantine adversaries

An insider can not be trusted to cooperate

Can not be trusted to correctly generate data (i.e. lie):

- Solution difficult to construct when
  - Many insider nodes collude
  - Not enough history is available
  - Single source of information

Can not be trusted to correctly deliver data:

- Solution difficult to construct when
  - Not enough adversary-free paths
  - Not enough redundancy
  - Correlated failures

Opportunities in multi-robot systems

- Cyber-physical enabled, mobile devices
  - Cyber-channels can be leveraged
  - Physics-based invariants
  - Adjust to provide needed redundancy
- Task specific predefined physical invariants
  - E.g. obstacles at certain locations
- In centralized architectures:
  - There is a point of trust and aggregation

In this talk

Consider two applications in two different scenarios

- **Multi-agent pathfinding** in a *centralized* setting
- **Consensus** in a *decentralized* setting (and applications based on it, tracking and localization)

Answer the following questions?

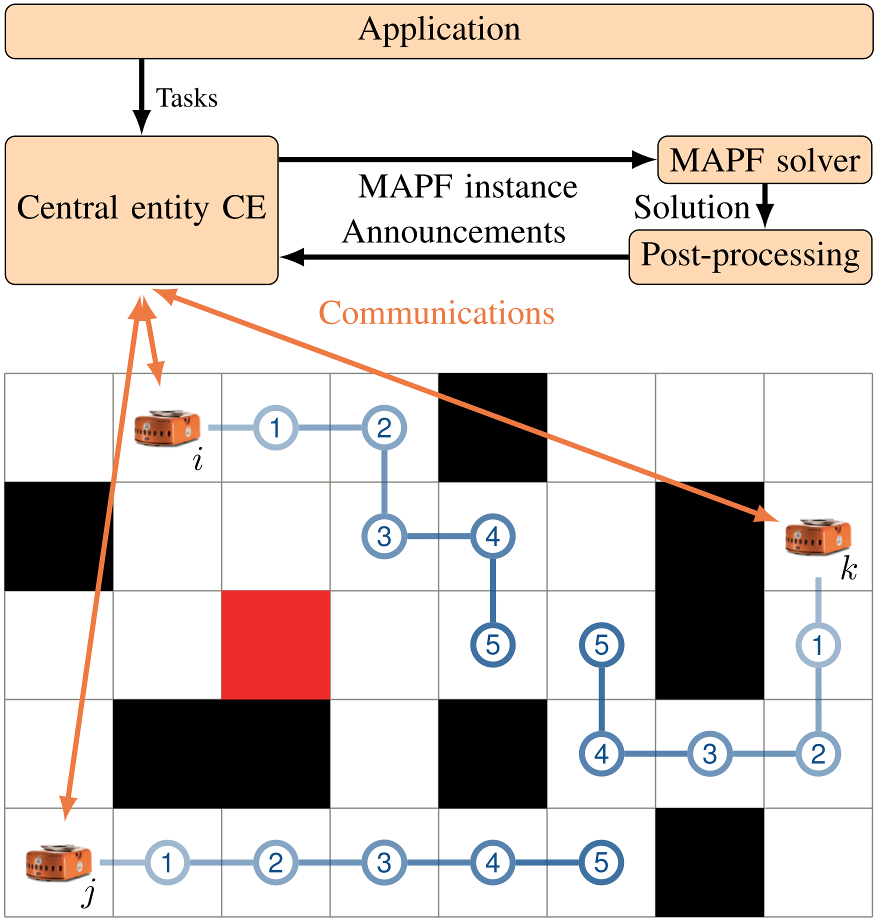
- What are relevant Byzantine attacks and what is their impact?
- How to design defenses (with provable security and scalable)?

Multi-agent pathfinding (MAPF)

- Automated planner that generates multi-robot plans
- Generating these *multi-agent motion plans* is known as multi-agent pathfinding (MAPF):
  - Structured or unstructured environments
  - Continuous or discrete robot dynamics
  - Under temporal specifications
  - Coordinated motions
  - Centralized or distributed decision making

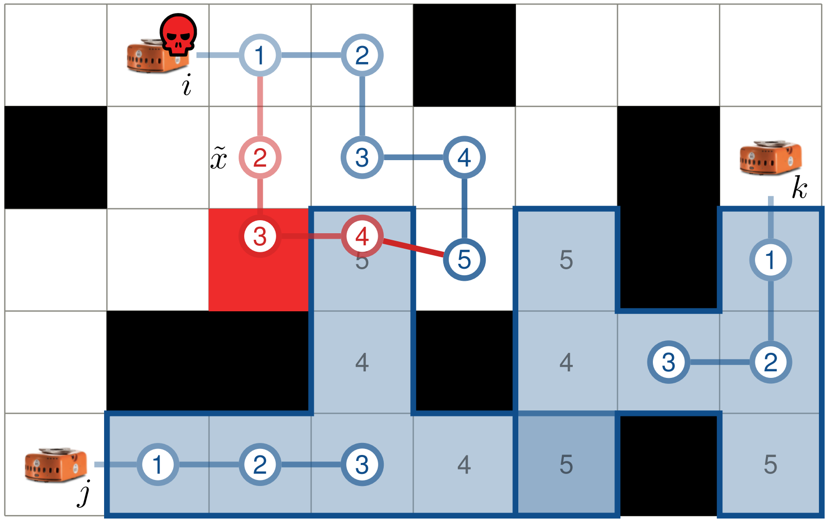
Automated warehouse

- Application generates requests for items to be fetched
- Motion plans consist of movement around and manipulation of different shelves for delivering the items to the target destinations.
- Map
- Obstacles
- Forbidden areas
- Central entity (CE)
- **Robots report their location periodically**



Plan-deviation attacks against MAPF

- Plan deviation:
  - Not follow the plan from the CE
- Forbidden plan deviation:
  - Get into forbidden areas
- Robots lie about their location
  - Cautious attacker (wants to stay undetected)
  - Bold attacker (does not care to stay undetected)



Solution overview

Attacker

- Moving toward the forbidden zone by leveraging the motion plan information received from the CE
- Trying to remain undetected by the CE by lying about its location

Defense

- Use robots to monitor other robots
  - CE computes *co-observation schedules* about the presence or absence of robots in certain locations at certain times
  - CE compares the reports from robots with the co-observations schedules to detect scenarios when compromised robots lied about their location
- Limit how much motion planning information the CE announces to the robots at any given time

Co-observation based detection

- For certain inputs, it is possible to compute the motion plan of the robots such that the resulting co-observation schedule has monitoring guarantees for plan-deviation attacks
- **Monotonicity of robot co-observations:** Co-observations increase the set of possible plan deviations that the CE can detect with respect to localization-based detection alone
- **Attack-Proof MAPF Plan**: Any forbidden deviation implies a change in the nominal observation schedule

Limitations of co-observation based detection

- Existence of an attack-proof plan
  - There is no guarantee that attack-proof MAPF plans exist for all MAPF instances
- Cost of the attack-proof plan
  - **Makespan**: time required for all robots to reach their respective goal locations (cost metric in MAPF planning)
  - When attack-proof MAPF plans exist, there is a trade-off between optimal-makespan MAPF plans and optimal-makespan attack-proof plans

Horizon-Limiting MAPF Announcements

Horizon limiting-announcements do not reveal enough information for the attacker to be certain that a given forbidden deviation will be undetected by the CE

- A CE that only makes horizon-limiting MAPF announcements maintains the security from cautious attackers that results from robot co-observations, but this time without the burden of computing attack-proof MAPF plans
- For bold attackers, no formal guarantee, we show experimentally, that bold attackers will have greater difficulty performing forbidden and undetected deviations as the information contained in the announcements decreases

In this talk

Consider two applications in two different scenarios

- **Multi-agent pathfinding** in a *centralized* setting
- **Consensus** in a *decentralized* setting (and applications based on it, tracking and localization)

Answer the following questions?

- What are relevant Byzantine attacks and what is their impact?
- How to design defenses (with provable security and scalable)?

Consensus for MSR

- **Linear Consensus Protocol**: the state of each agent is updated based on a linear combination of its own state and the states of its neighbors
- **Weighted-Mean Subsequence Reduced Algorithm**
  - Designed to tolerate F byzantine robots
  - Discards the F highest and F lowest values, then use LCP
  - To converge requires the connectivity graph of the robots to be at least (2F +1)-vertex-connected. Difficult to achieve in practice.

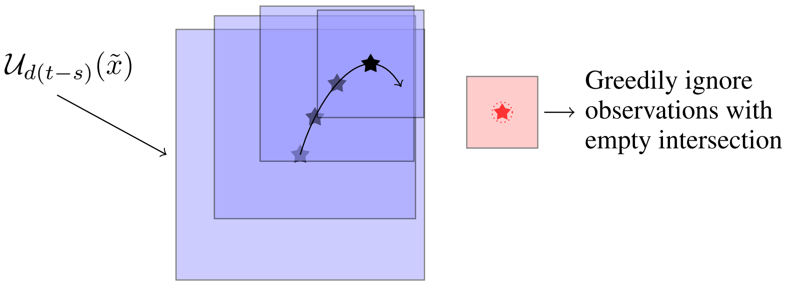
$$x_i(t) = \sum_{j \in \mathcal{N}_G(i)} \alpha_j x_j(t-1) \text{ where } \sum \alpha_j = 1$$

Decentralized Blocklist Protocol

- **Main idea**: Based on locally-made observations, cooperative robots accuse misbehaving peers. The accusations propagate through the network via flooding and are used as input to a matching algorithm that outputs a blocklist
- The precise rules used to decide if and when an accusation should be issued are application-specific
- Each robot locally maintains a set of accusations that it has received. A subset will be locally computed by using any deterministic maximum matching algorithm (such as Edmond's) to form the blocklist
- **"i accusing j " can be understood as "i is Byzantine or j is Byzantine (or both are)."**

Applications: Target Tracking

- Goal: robots locate and cooperatively follow a mobile target that has a maximum speed
  - In each timestep, robots sort received observation messages by observation time, and choose the most recent one to transmit to its neighbors.
- Controller:
  - Compute a heading vector pointing to the target from their current location and move towards the target.
  - Robots that do not directly observe the target rely on received observation messages to compute their heading vector.



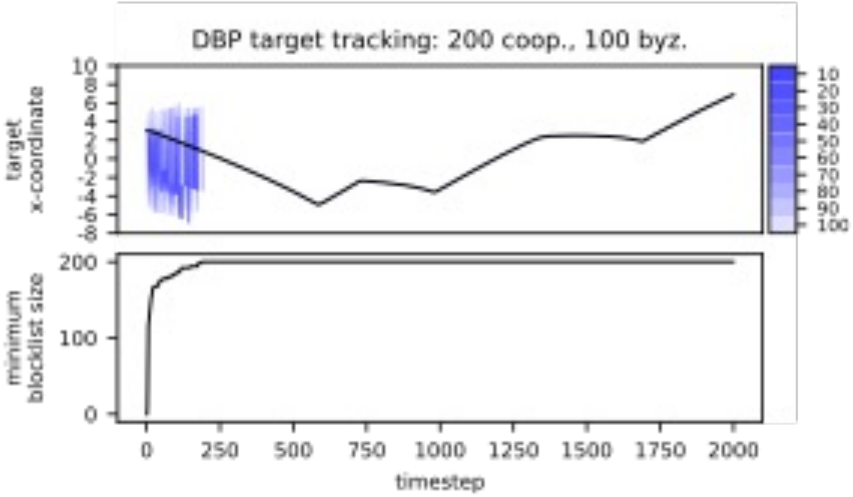
Greedily ignore observations with empty intersection

Accusation rules:

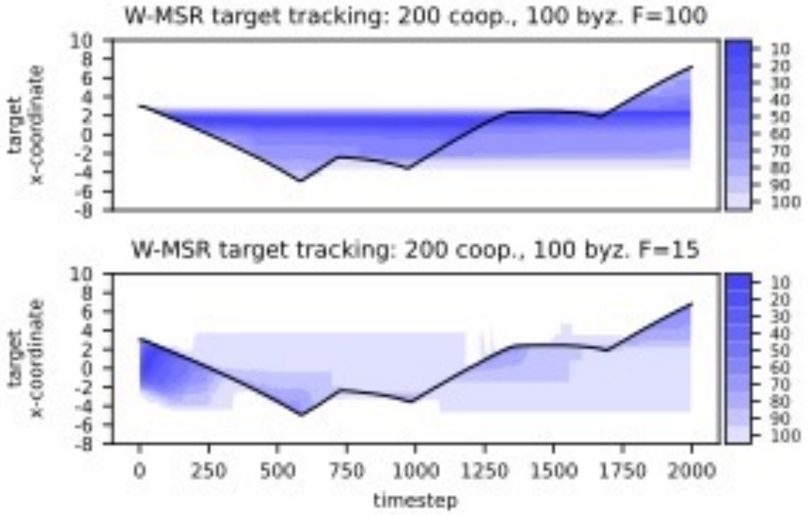
- Observations can not travel faster-than-physically-possible
- Missed an observation that it should have made if the received observation was legitimate
- The target couldn't possibly have moved fast enough from the received observation location to the place where it observed it presently
- Detects oscillations from a single observer

Target tracking: Experiments

- DBP-based target tracking performance.
- At timestep ~ 200, all Byzantine robots have been blocked on each honest robot, and the honest robots track the target with close to no error



- W-MSR-based target tracking performance.
  - F = 100 guarantee safety, the information about the moving target cannot propagate through the cooperative robots
  - F = 15 has no safety guarantee but allows a subset of the robots to track the target successfully. However, the influence of the Byzantines is never removed.



Summary

- **Multi-agent pathfinding** in a *centralized* setting
  - Horizon-limited co-observation that exploited the characteristics of the task (given map, obstacles, position of forbidden areas) to issue incremental plan to the robots, plans that are guaranteed to prevent cautious attackers and limit bold attackers
- **Consensus** in a *decentralized* setting
  - Show how to build a decentralized blocking protocol that leverages application-specific rules to generate accusations such that all robots can compute the maximum list of such accusations
  - Better scalability and less strict connectivity requirements than state of the art WMSR

Acknowledgements and Publications

- Joint work with Kacper Wardega,Roberto Tron, and Wenchao Lu from Boston University and my PhD student Max von Hippel
- **HoLA Robots: Mitigating Plan-Deviation Attacks in Multi-Robot Systems with Co-Observations and Horizon-Limiting Announcements.** Kacper Wardega, Max von Hippel, Roberto Tron, Cristina Nita-Rotaru and Wenchao Li. The 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS), May 2023.
- **Byzantine Resilience at Swarm Scale: A Decentralized Blocklist from Inter-robot Accusations.** Kacper Wardega, Max von Hippel, Roberto Tron, Cristina Nita-Rotaru and Wenchao Li. The 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS), May 2023.