Heterogeneous Multi-Robot Adversarial Patrolling Using Polymatrix Games

Alexander Langley, Vikas Dhiman and Henrik Christensen

University of California San Diego, CA, USA

{alangley,vdhiman,hichristensen}@ucsd.edu

Abstract. We consider the problem of patrolling an area with a heterogeneous multi-robot team. The goal for the patrolling robots is to detect intrusions while minimizing the probability of being detected. The heterogeneous teaming problem has been widely studied using coordination strategies such as minimax and strategic alliances. In most cases a centralized coordination is assumed. We present an approach based on a zero-sum polymatrix approach and derive its basic characteristics. In polymatrix games a patroller consider other team members' possible strategies when formulating their own strategy, while eliminating the requirement of considering the joint action space. We systematically evaluate our approach in randomly generated gridworld environments, and also demonstrate the performance of the approach in a large scale realistic environment. The results clearly demonstrate that the polymatrix approach has improved detection rates without an increase in complexity.

1. Introduction

Multi-robot deployment for security applications is becoming pervasive. We see multi-robot teams used for monitoring forest fires [1], for border control inspection [2], and for monitoring infrastructure [3] and bodies of water such as lakes [4]. The deployment of multiple autonomous vehicles can be formulated as a patrolling problem, where a multi-robot team of patrollers seeks to detect intruders in a large area [5, 6]. Patrolling, like the closely related Perimeter Defence [7] and Pursuit-Evasion [8, 9] problems, has applications in domains such as military operations [3], search and rescue [10], and checking in public transportation systems [11].

In this paper, we are primarily interested in a scenario in which a heterogeneous multi-robot team must provide reconnaissance of a large area in which intrusion has already occurred. The intruding robot's mission is to occupy and surveil the environment, but its exact type (ground or aerial) and position are not known to the patrolling team. The patrolling robots' goal in this situation is twofold: they seek to detect the intruder, but they also want to minimize the chance that they will be detected themselves.

To this end, we envision a team of ground and aerial robots that can patrol large-scale environments. In large-scale environments, the robots may not be able to maintain connectivity with each other at all times, which makes decentralization a necessity. However, most works in the literature model adversarial patrolling as a two-player game, which cannot be easily implemented in a decentralized fashion.

To address this limitation, we propose the use of polymatrix games [12] for adversarial patrolling. Polymatrix games allow us to model each patroller as a separate player who formulates their strategy

[0] This is an open-access version of the article corresponding to the publication https://doi.org/10.1007/978-3-031-09909-0_2

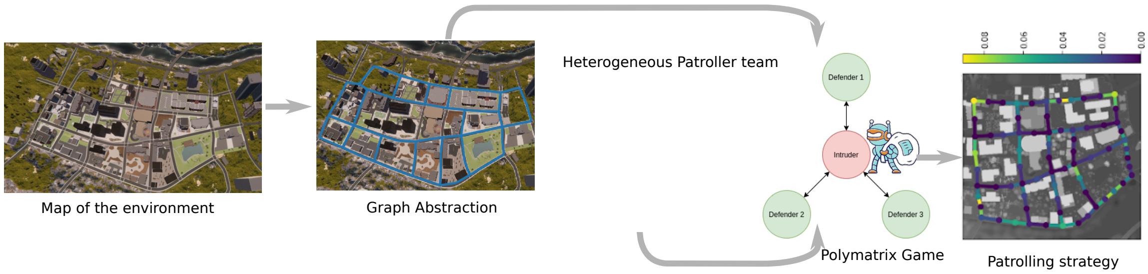

Figure 1: Flowchart for generating adversarial patrolling strategies using Polymatrix games.

given information about the other robots on their team. This formulation has the potential to be implemented as a decentralized algorithm on each robot. In this work, we present and evaluate a centralized version of the algorithm, leaving the decentralized version for future work. The main objective is to demonstrate the value of polymatrix games and for that the increased complexity of distributed coordination is not required.

On the other extreme, the polymatrix game is superior to a näive solution of playing pairwise independent minimax games. Patrolling using polymatrix games accounts for the possible strategies of other patrollers, which is important because the exact composition of the patrolling team affects the intruder's best response. Without considering the possible strategies of their teammates, patrolling robots will make less informed predictions about how intruders will behave. Hence, our approach is a step toward decentralized agents who are also able to reason about the possible actions of their teammates.

Figure 1 summarizes our approach: Given an environment, we extract a graph representation. A team of autonomous robots comprised of ground and aerial robots then plays a polymatrix game with the intruder to decide where each agent will patrol and how often.

The main *contribution* of this work is to apply the polymatrix game formulation to the problem of multi-robot adversarial patrolling with a heterogeneous team of robots. To the best of our knowledge, our work is the first to apply zero-sum polymatrix games to the adversarial patrolling problem.

The rest of this paper is organized as follows: In Section 3, we discuss our problem formulation. In Section 4, we introduce polymatrix games, discuss their application to multi-robot patrolling, and show how to compute patrolling strategies by finding their Nash equilibria. In Section 7, we evaluate our approach experimentally in randomly generated gridworld environments. In Section 8, we discuss the application of our approach to a realistic, large-scale environment.

2. Related work

Patrolling is generally classified as either Regular or Adversarial patrolling [6]. Regular patrolling [5] aims to periodically visit important locations so that the duration between visits to locations is minimized. Regular patrolling, however, is typically deterministic. In the presence of an adversary which is able to observe the patrollers' behavior before deciding on a plan for intrusion, deterministic patrolling allows an intruder to invade a location when it knows that patrollers will be elsewhere. Adversarial patrolling addresses this problem by using stochastic strategies that resist the intruder's ability to learn and predict the patrollers' actions [6].

Adversarial patrolling is often formulated and solved in the literature as a two-player Bayesian Stackelberg game [6, 13–15]. In a Stackelberg game the leader (in the case of adversarial patrolling, the patroller team) chooses a strategy and the follower (here an intruder) observes and adapts to the leader's strategy. Using Stackelberg game to model patrolling accounts for the intruder's ability to observe the patrollers' behavior and formulate an intrusion plan which exploits weak points in the patrollers' strategies. We observe in our scenario, though, that Stackelberg game strategies are overly conservative. In large scale patrolling scenarios, the intruders cannot observe patrollers strategies without being in the environment. Moreover, in dynamic decentralized situations where the patrollers are recomputing their patrolling strategies on the fly when given new information about the state of their team, the intruders have little leverage to observe and learn patrollers' strategies.

A similar line of work [11, 16] models the patrolling game with multiple intruders as a two-player game in which one patrolling player selects strategies in the patrolling robots' joint action space. Since the patrollers' joint strategy space grows exponentially in the number of agents, a common simplifying assumption is that of linearly separable utilities, which decouples the patrolling robots' strategies by considering agents which can maximize their payoffs independently of one another. In [17], the authors use this assumption to justify a treatment of the multi-patroller game in which patrolling strategies can be computed by finding a minimax strategy for each patroller individually.

For a comprehensive survey of the related work, we direct the interested reader to [5,6].

3. Background

As already mentioned in the introduction, there are a number of approaches to doing multi-agent/player coordination. The simplest strategy is to use a random coordination strategy, which makes no assumption about intruder behavior or mission. Another alternative is an independent minimax strategy where each patroller can achieve at least their individual worst-case outcome without building a coalition. Finally, we have our proposed polymatrix approach which is explained in the next section.

3.1. Polymatrix games

The polymatrix game formalism provides a useful abstraction for describing a network of pairwise interactions between players, where the nodes in the network correspond to players, and the edges represent the pairwise interactions between them. A polymatrix game $\mathscr{G} = (\mathscr{V}, \mathscr{E}, \mathscr{S}, \mathscr{U})$ is defined as [12]:

- A finite set $\mathscr{V} = \{1, \ldots, n\}$ which correspond to vertices (players) in the interaction graph.
- A finite set $\mathscr{E}$ of edges connecting players in $\mathscr{V}$, denoting a game-interaction between the connected players.
- A set of strategies $\{S_i\}_{i=1}^n$ for each player, which defines the set of strategy profiles $\mathscr{S} = \prod_{i=1}^n S_i$.
- For each edge $(i, j) \in \mathscr{E}$, a corresponding two-player game defined by (u^{ij}, u^{ji}), where $u^{ij} : S_i \times S_j \to \mathbb{R}$ is the payoff function for player i in the game it plays with player j. Let $\mathscr{U} = \{u^{ij}\}_{(i,j)\in\mathscr{E}}$ be the set of all the pairwise-payoff functions.

For each strategy profile $\mathbf{s} \in \mathscr{S}$, the total utility to player i is given by the sum over their neighbor's utilities $u_i(\mathbf{s}) = \sum_{j\in\mathrm{Nbr}(i)} u^{ij}(s_i, s_j)$, where $\mathrm{Nbr}(i) = \{j : (i, j) \in \mathscr{E}\}$ denotes the neighbors of the node i in the graph.

If for all strategy profiles $\mathbf{s} \in \mathbf{S}$, we have $\sum_{i\in\mathscr{V}} u_i(\mathbf{s}) = 0$, then we say that $\mathscr{G}$ is *zero-sum*. If we have that for all $(i, j) \in \mathscr{E}$ and $\mathbf{s} \in \mathbf{S}$, $u^{ij}(s_i, s_j) + u^{ji}(s_j, s_i) = 0$, then we say that $\mathscr{G}$ is *pairwise zero-sum*. Note that each player must commit to a single strategy to play in all games in which it participates.

In [12], the authors show that we can compute a Nash equilibrium strategy profile for a zero-sum polymatrix game $\mathscr{G}$ by solving the linear program

$$\min_{\{\mathbf{y}_i, w_i\}_{i=1}^n} \sum_{i\in\mathscr{V}} w_i, \qquad \text{s.t.} \quad w_i \geq u_i(s_i, \mathbf{y}_{-i}), \quad \forall i \in \mathscr{V}, \forall s_i \in S_i, \qquad \text{and } \mathbf{y}_i \in \Delta(S_i), \quad \forall i \in \mathscr{V} \tag{1}$$

where $\mathbf{y}_{-i}$ denotes the mixed strategies of players in $\mathscr{V} \setminus \{i\}$ and $\Delta(S_i)$ is the set of probability distributions over S_i. They also show that the optimal objective value of (1) is zero, and that a Nash equilibrium is obtained at this value. If $\mathscr{G}$ is pairwise zero-sum, then each player has a unique payoff that it receives at any Nash equilibrium, known as its *value in the game* [18].

The most salient difference between the Linear Program in (1) and an independent minimax formulation is that, rather than optimize a lower bound on their own worst case payoff, each player seeks to place an upper on the payoffs of the players with whom they compete.

4. Problem formulation

We are interested in best-effort patrolling of a given area to defend it from adversarial intruders, which we will do by deploying a team of n robots. Each patrolling robot takes on one type from a set of robot types $\Lambda = \{Ground, Aerial\}$. The intruder may appear as either a ground or aerial robot. Aerial robots have an advantage with respect to their detection range and can detect intruders at greater distances than ground units. We also consider, though, that they are vulnerable to ground units: Aerial robots are averse to coming into the detection range of ground units, since ground units may be able to attack aerial robots in a real patrolling scenario.

Let the patrolling area be approximated by a directed road network $\mathscr{R} = (\mathscr{N}, \mathscr{T})$ where $\mathscr{N} = \{\mathbf{p}_1, \ldots, \mathbf{p}_{|\mathscr{N}|}\}$ are the nodes (road junctions) and $\mathscr{T}$ is the set of possible transitions (directed road segments) between nodes. Each patroller's strategy set consists of the transitions in $\mathscr{T}$. We interpret the mixed strategies $\{\mathbf{x}^{(i)}\}_{i=1}^{n}$ as the probabilities that patrolling agents choose to patrol along each road segment in $\mathscr{R}$. The goal of the patrollers is to detect intruders, who attempt to go undetected while occupying road segments in $\mathscr{R}$. Patrollers also want, however, to avoid detection by intruders as much as possible. Otherwise, patrolling robots risk confrontations with adversarial units who can damage them. Thus, unlike typical adversarial patrolling, we assume that both patrollers and intruders want to avoid detection by their opponents.

Let $u_p, u_a : \mathscr{T}^{n+1} \times \Lambda \to \mathbb{R}$ represent the utility for the patrolling team and the intruder, respectively. The payoffs are dependent upon the transitions in $\mathscr{T}$ chosen by the patrollers and the intruder, as well as the robot type used by the intruder. We assume that the game played between the patrolling team and the intruder is zero-sum i.e. $u_d = -u_a$. We also assume that the patrollers' utilities are *linearly separable* [17]. This means that there exist utility functions $u^{(i)} : \mathscr{T}^2 \times \Lambda \to \mathbb{R}$, one for each patroller, such that $u_p(t_1, \ldots, t_n, t_a, \lambda_a) = \sum_{i=1}^{n} u^{(i)}(t_i, t_a, \lambda_a)$, where $\{t_i\}_{i=1}^{n}$ and t_a are the transitions in $\mathscr{T}$ chosen by the patrollers and intruder, respectively, and λ_a is the robot type chosen by the intruder.

In order to ensure that the probability of an agent transition to a particular node remains consistent, we enforce that the probability of transitioning into a node must be equal to the probability of transitioning out of the node,

$$\mathbf{x}^{(i)} \geq \mathbf{0}, \quad \sum_{t \in \mathscr{T}} x_t^{(i)} = 1 \quad \text{and} \sum_{t \in \text{In}(v)} x_t^{(i)} = \sum_{t \in \text{Out}(v)} x_t^{(i)} \quad \forall v \in \mathscr{N}; \qquad \forall i \in \{1, \ldots, n\}, \tag{2}$$

where $x_t^{(i)}$ denotes the tth element of the mixed-strategy vector $\mathbf{x}^{(i)}$, $\text{In}(v)$ and $\text{Out}(v)$ denote the incoming and outgoing transition for a node, respectively. Similarly, we denote $\mathbf{y} \in \Delta(\mathscr{T} \times \Lambda)$ the mixed strategy for the intruder in choosing the road segments to traverse as well as the intruder type. We enforce a similar constraint for a valid intruder strategy $\mathbf{y}$,

$$\mathbf{y} \geq \mathbf{0}, \quad \sum_{\lambda \in \Lambda} \sum_{t \in \mathscr{T}} y_{t,\lambda} = 1, \qquad \text{and} \sum_{t \in \text{In}(v)} y_{t,\lambda} = \sum_{t \in \text{Out}(v)} y_{t,\lambda} \quad \forall v \in \mathscr{N}, \lambda \in \Lambda. \tag{3}$$

Formally, we model the patrolling problem as a multi-player game where patrollers try to maximize their utility by choosing an appropriate mixed-strategy $\{\mathbf{x}^{(i)}\}_{i=1}^{n}$ while the intruder tries to minimize the patroller tries to minimize the same,

$$\max_{\{\mathbf{x}^{(i)}\}_{i=1}^{n}} \min_{\mathbf{y}} \sum_{i=1}^{n} u^{(i)}(\mathbf{x}^{(i)}, \mathbf{y}), \qquad \text{s.t. (2) and (3).} \tag{4}$$

5. Utility Model

The patroller's utility is dependent on a detection event. A patrolling robot at position $\mathbf{p}_d$ is said to detect an attacker at position $\mathbf{p}_a$ when the attacker comes within the detection range r_d of the patrolling robot.

$$\delta(\mathbf{p}_d \to \mathbf{p}_a) = \begin{cases} 1 & \text{if } \|\mathbf{p}_d - \mathbf{p}_a\|_2 < r_d \text{ and } \mathbf{p}_a \text{ is in line of sight from } \mathbf{p}_d \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

The detection is said to be mutual when both $\mathbf{p}_d$ and $\mathbf{p}_a$ are within each other's detection range $\delta(\mathbf{p}_d \rightarrow \mathbf{p}_a) = \delta(\mathbf{p}_a \rightarrow \mathbf{p}_d) = 1$.

We model two kinds of utility, occupation utility $u_{\text{occ}}(\mathbf{p}, \lambda) \in \mathbb{R}^+$ and detection utility $u_{\text{det}} \in \mathbb{R}^+$. In the absence of a detection event, the patrollers get a utility of $-u_{\text{occ}}(\mathbf{p}_a, \lambda_a)$ while the intruder gets $+u_{\text{occ}}(\mathbf{p}_a, \lambda_a)$. Whenever, a detection event occurs the detection utility is determined by three factors, (1) whether the detection is mutual (both players detect each other) or asymmetric (2) robot type (ground or aerial) and (3) team (patroller or intruder). In case of a mutual detection, the Ground robot has the advantage against the Aerial robot, hence the patroller utility is positive if its robot type is Ground.

$$u^{(d)}_{\text{mutdet}}(\lambda_d, \lambda_a) = \begin{cases} u_{\text{det}} & \text{if } \lambda_d = \text{Ground}, \lambda_a = \text{Aerial} \\ -u_{\text{det}} & \text{if } \lambda_d = \text{Aerial}, \lambda_a = \text{Ground} \\ 0 & \text{otherwise} . \end{cases} \tag{6}$$

On the other hand, in case of an asymmetric detection, the detector gets the u_{det}, while the detected gets $-u_{\text{det}}$. When the detector is an intruder the patroller gets $-u_{\text{det}} - u_{\text{occ}}(\mathbf{p}_a, \lambda_a)$,

$$u^{(d)}_{\text{pos}}(\mathbf{p}_d, \mathbf{p}_a, \lambda_a) = \begin{cases} u^{(d)}_{\text{mutdet}}(\lambda_d, \lambda_a) & \text{if } \delta(\mathbf{p}_d \rightarrow \mathbf{p}_a) \wedge \delta(\mathbf{p}_a \rightarrow \mathbf{p}_d) \\ -u_{\text{det}} - u_{\text{occ}}(\mathbf{p}_a, \lambda_a) & \text{if } \delta(\mathbf{p}_a \rightarrow \mathbf{p}_d) \wedge \neg\delta(\mathbf{p}_d \rightarrow \mathbf{p}_a) \\ u_{\text{det}} & \text{if } \delta(\mathbf{p}_d \rightarrow \mathbf{p}_a) \wedge \neg\delta(\mathbf{p}_a \rightarrow \mathbf{p}_d) \\ -u_{\text{occ}}(\mathbf{p}_a, \lambda_a) & \text{otherwise} . \end{cases} \tag{7}$$

Now that we have the utilities for the positions of the robots, we can define utilities for transitions over road segments. Assuming a constant velocity model, the position is parameterized over the road segment using normalized time $\rho \in (0, 1)$. The position of the patroller at any time ρ is given by $\mathbf{p}_d(\rho) = (1 - \rho)\text{start}(t_d) + \rho\text{end}(t_d)$. The utility over road segments $t_d, t_a \in \mathscr{T}$ is computed by integrating the positional utility over the linear road segment.

$$u^{(d)}(t_d, t_a, \lambda_a) = \int_0^1 u^{(d)}_{\text{pos}}(\mathbf{p}_d(\rho), \mathbf{p}_a(\rho), \lambda_a) d\rho. \tag{8}$$

Note that utility computation has to be done only once for a static map and a particular robot type.

6. Game theoretic patrolling

We consider two approaches, each with a different set of assumptions to address the proposed problem 1) Independent Minimax Approach and 2) Polymatrix Approach.

6.1. Independent Minimax Approach

A näive approach to solving the multi-patroller problem would be to play a minimax game in the joint strategy space of the patrollers. This approach, however, is not feasible with even a small number of patrollers. We follow an alternative approach where each patroller is modeled as an independent player that plays a minimax game against a single intruder. This approach scales linearly with the number of intruders. Moreover, each patroller-intruder game becomes a two-player zero-sum game which can be efficiently solved with a linear program:

$$\max_{\{\mathbf{x}_i\}_{i=1}^n} \quad z_i, \qquad \text{s.t} \quad z_i \mathbf{1} \leq \mathbf{x}^{(i)\top} \mathbf{U}^{(i)}, \text{ and (2)}, \tag{9}$$

where $z_i \in \mathbb{R}$ is the lower bound on the ith patroller's utility given the possible actions taken by the intruder. Here, $\mathbf{U}^{(i)} = [[u^{(i)}(t_i, t_a, \lambda)]_{t_i \in \mathscr{T}}]_{t_a \in \mathscr{T}, \lambda \in \Lambda}$ is the utility matrix for the ith patroller.

6.2. Polymatrix approach

We propose the use of polymatrix games [12] for the task of adversarial patrolling. We instantiate the polymatrix game $\mathscr{G} = \{\mathscr{V}, \mathscr{E}, \mathscr{S}, \mathscr{U}\}$ for adversarial patrolling as follows

- Each robot, the n patrollers and an intruder are modelled as separate players, i.e. $\mathscr{V} = \{1,\ldots,n+1\}$, where $n+1$ is the intruder.
- We define a set of *pairwise zero-sum* games between each patroller and the intruder as $\mathscr{E} = \{(1,n+1),(2,n+1),\ldots,(n,n+1)\}$.
- The mixed strategy of each patroller is a probability distribution over all edges of the road network $\mathscr{R}$, i.e. $S_i = \mathscr{T}$ for all $i \in \{1,\ldots,n\}$. The intruder also chooses its robot type, i.e. $S_{n+1} = \mathscr{T} \times \Lambda$.
- The utilities are as modeled in Section 5, so that a set of Utility matrices $\mathbf{U}^{(i)} = [[u^{(i)}(t_i,t_a,\lambda)]_{t_i=1}^{\mathscr{T}}]_{t_a=1,\lambda\in\Lambda}^{\mathscr{T}}$ define the payoff for each patroller.

Such a polymatrix formulation can be solved to obtain the Nash equilibrium as a linear program:

$$\min_{\mathbf{y},\{\mathbf{x}\}_{i=1}^{n}} \quad \sum_{i=1}^{n} z_i + w, \qquad \text{s.t. } w\mathbf{1} \geq -\sum_{i=1}^{n} \mathbf{x}^{(i)\top}\mathbf{U}^{(i)}, \quad z_i\mathbf{1}^\top \geq \mathbf{U}^{(i)}\mathbf{y} \qquad \forall i \in \{1,\ldots,n\}, (2), \text{and } (3), a1 \tag{10}$$

where z_i is the upper bound on the ith patroller's utility and w is the upper bound on the intruder's utility. Even though solving this problem finds an approximate, and not an exact, Nash equilibrium, due to the extra constraints on movement, we show in Section 7 that our method still performs well despite the intruder being allowed to exploit the gap in utility caused by the error.

7. Experimental evaluation

In this section, we compare the performance of our approach using polymatrix games to patrolling strategies obtainted using independent minimax strategies and uniformly random patrolling strategies.

Intruder strategy optimization We are interested in comparing the worst case scenario for the patrolling team given that they follow strategies prescribed by each of the three methods. Hence, to evaluate the performance of each of the three patrolling approaches, we compute the intruder's strategy by finding a best response given the patrollers' strategies. Specifically, given patroller strategies $\{\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(n)}\}$, we solve the problem,

$$\max_{\mathbf{y}} \quad \sum_{i=1}^{n} \mathbf{x}^{(i)\top}\mathbf{U}^{(i)}\mathbf{y} \qquad \text{s.t} \quad (3). \tag{11}$$

7.1. Experimental setup

We use 100 randomly generated gridworld environments to validate our computed patrolling strategies, of which Figure 2 shows five examples. The environments were generated by starting with a 10×10 grid of vertices and randomly removing vertices with a probability of 0.35. After that, vertices not connected to the vertex at the point (0, 0) were also removed to ensure that all the generated graphs are connected.

In our experiments, we interpret white space between vertices as buildings through which robots can not detect each other. We assume that quadrotors fly below the tops of buildings in order to avoid making themselves too highly visible to adversaries. Hence, all robots can detect other robots only if they are located at vertices which are connected to their current location by an unbroken, straight path and are within their sensing range. In our simulations, ground rpbpts' sensing range extends to adjacent vertices, while aerial robots may detect other robots up to 3 edges away.

We run each simulation for 20 time steps, where patrollers and intruders spawn at locations according to estimated strategies. At every time step, both the patrollers and the intruder sample a nearby edge to move from their respective optimal strategies. We evaluate each method based on the proportion of intruders detected, the number of times a patroller is detected by an intruder, and the average time taken to detect each intruder. We present results for two different patroller teams, each computing strategies using $u_{occ} = 1$ and $u_{occ} = 2$. By varying the value of u_{occ}, we compare the results when facing an intruder which either has no preferences between occupation and detecting patrollers, or which prioritizes occupying

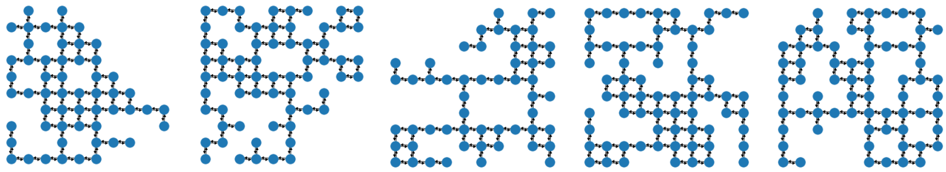

Figure 2: Examples from randomly generated gridworld environments.

territory without detection, respectively. The first team consists of two ground robots and one aerial robot, while the second team consists of one ground robot and two aerial robots. For each of the 100 environments, we run 10,000 simulations for both values of u_{occ}, each with a duration of 20 time steps. In order to reduce the time required to compute the utility matrix, we test for detection between two robots only at vertices in the road network rather than integrating over the road segment (8).

7.2. Executing computed strategies

At the beginning of each simulation, the ith patrolling robot is initialized by sampling an edge from its mixed strategy $\mathbf{x}^{(i)}$ and placing the patroller on the terminal vertex of that edge. At each timestep of the simulation the ith patroller chooses a transition in $\mathrm{Out}(k)$ by sampling from the distribution defined by $\mathbb{P}(t) = \frac{x_t^{(i)}}{\sum_{t\in \mathrm{Out}(k)} x_t^{(i)}}$.

At any time during the simulation, a new intruder has a probability of $p = 0.3$ of entering the environment. New intruders are initialized by first drawing from the distribution $\mathbb{P}(\lambda) = \sum_{t\in\mathscr{T}} \mathbf{y}(t,\lambda)$ to determine the robot type as which they will instantiate themselves. The intruder's position is then initialized in a way analogous to the patrollers. An intruder may not change its type after entering the environment. Once an intruder is detected by a patroller, that intruder is considered to be neutralized and has no effect on the rest of the simulation (cannot be detected again and cannot detect patrollers).

7.3. Simulation results

Quantitative simulation results are shown in Figure 3 for the two patrolling team configurations and the parameters $u_{occ} = 1$ and $u_{occ} = 2$.

We see that our approach using polymatrix games is superior to the baseline method with respect to the proportion of intruders detected in each simulation, regardless of team composition and value of u_{occ}. This demonstrates that the improvement in the rate of intruder detection our method provides is robust to changes in model parameters and team composition. With respect to the time for which intruders were allowed to occupy the road network without detection, the polymatrix and individual minimax approaches are the same, but both methods perform better than random patrolling strategies on this metric. Recall also that we measure the time-to-detection based on the intruders that were detected in the first place. This means that robots using polymatrix strategies are able to detect more intruders, just as quickly as when they use their independent minimax strategies.

We see also, however, that patrolling agents using our polymatrix strategies are more likely to be detected by intruders. This reflects the inherent tradeoff between detecting intruders and avoiding detection by them. We observe that patrollers who detect more intruders are more likely to come within detection range of those intruders. Our results also show, however, that this is a reasonable tradeoff to make: the increase in performance in terms of the proportion of intruders detected when patrollers use their polymatrix strategies greatly outweighs the increase in their likelihoods of being detected.

8. Application to a realistic environment

In this section, we show how to adapt our approach to the problem of multi-robot patrolling in a large-scale, realistic simulated environment. We use a custom Unity based simulator in which robots can be spawned and moved around with realistic physics and graphics (See Figure 4). We improve upon

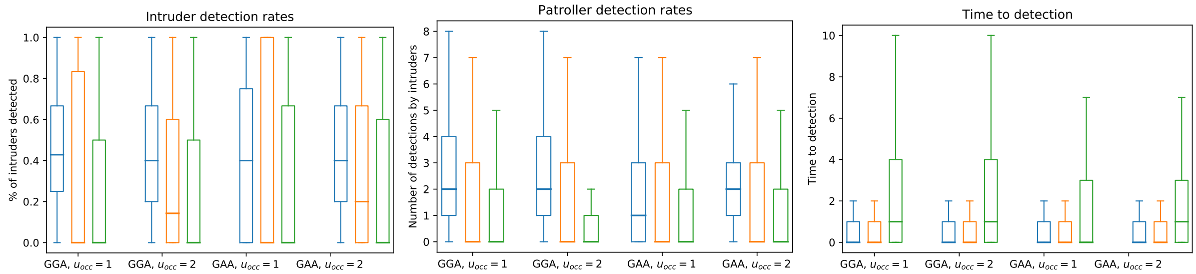

Figure 3: Simulation results for different team compositions and values of u_{occ}. GGA denotes a patroller team composed of two ground and one aerial robot, while GAA denotes a team of one ground and two aerial robots. The results for the polymatrix, independent minimax, and random patrolling strategies are blue, orange, and green, respectively. We see that *our* approach using polymatrix games is superior to the baseline method with respect to the proportion of intruders detected in each simulation, although with a tradeoff of marginal increase in patrollers detected.

the realism of Section 5 by incorporating different movement speeds for each robot type (aerial robots move more quickly than ground robots). We also use a more sophisticated model of visibility between points and consideration of the terrain complexity, as well as a measure of a robot's range of visibility at each position, along each road segment. In this section, we consider an intruder who wants to occupy road segments along which it has high visibility but which have low terrain complexity. In other words, intruders seek to maximize the extent to which they can survey the surrounding area while minimizing the cost of traveling through the environment.

We assume in this section that we are given maps assigning terrain complexity and visibility scores to each point in the environment for each robot type. We then compute the occupation utility for the intruder at a point $\mathbf{p}$ by $u_{occ}(\mathbf{p},\lambda) = u_{vis}(\mathbf{p},\lambda) - u_{terr}(\mathbf{p},\lambda)$, where u_{vis} and u_{terr} are mappings assigning a visibility score and terrain complexity to each point and robot type (See Fig. 4). We then normalize the occupation utilities to the range $[0,1]$.

8.1. Qualitative analysis of patrolling strategies

Figure 5 shows visualizations of the strategies chosen by the patrollers and the intrusion plan chosen by the intruder under the Nash equilibrium strategy profile. Note that the two ground units choose the same strategies, so we visualize a strategy for only one of them. The edge colors indicate the probability that a robot chooses to move along the corresponding road segment in either direction. The vertex colors indicate the probability of staying in the same location. The intruder strategies shown visualize the strategy used by the intruder when choosing to appear as either a ground or aerial robot.

Here, the intruder chooses to appear as an aerial robot with a probability of 0.89. Choosing to send mostly aerial robots in this environment makes sense, since they have greater visibility everywhere in the environment than ground robots do and can move at greater speeds. This allows the intruder to maximize the utility it receives from occupying the road network without being detected. Additionally, using ground robots is less advantageous for the intruder since the patrolling team has more aerial robots than ground robots: Recall that the primary advantage of ground robots is that they receive a high payoff by getting sufficiently close to aerial robots. Finally, aerial robots can more easily obtian high occupation utilities, since they have low or no terrain complexity everywhere and high visibility relative to ground robots.

A qualitative analysis of the strategies shown in Figure 5 shows that they exhibit many intuitive features. We see that patrolling ground units are likely to patrol along road segments where they can detect intruding aerial robots. We see also a division of labor between patrolling ground and aerial robots: Ground robots patrol in areas along the perimeter of the road network, while aerial robots patrol the interior of the urban environment. This is because aerial units can fly over and see above smaller buildings, making them more suitable to detect intruders who attempt to hide between buildings.

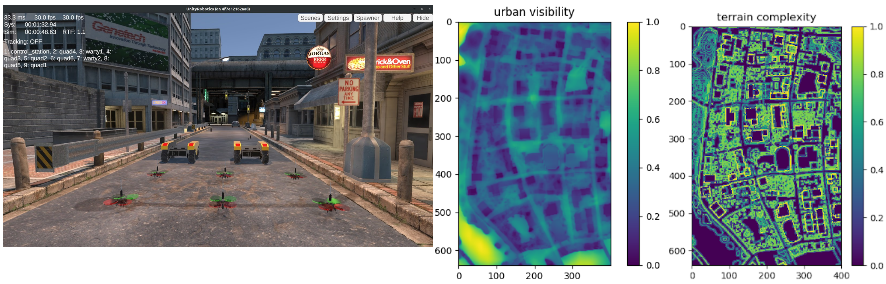

Figure 4: From left to right: (1) First person view of the unity based realistic simulator in which the patrolling experiments were done. (2) Visibility heatmap for the patrolling region used as $u_{\text{vis}}(\mathbf{p}, \lambda)$. (3) Terrain complexity heatmap for the patrolling environment in the simulator used as $u_{\text{terr}}(\mathbf{p}, \lambda)$.

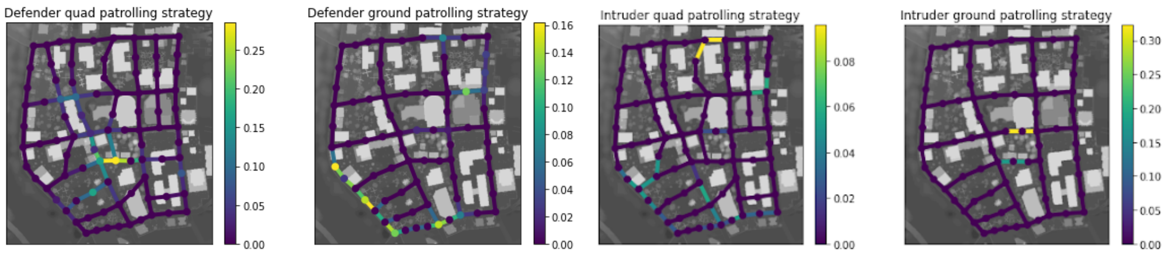

Figure 5: In order from left to right: Patrolling aerial robot strategy, patrolling ground robot strategy, intruding aerial robot strategy, intruding ground robot strategy.

Intruding aerial robots, on the other hand, attempt to occupy the perimeter of the environment. One reason for this is that they can obtain the most utility for occupying the bottom-left region of the environment. This is a highly valuable area of the environment for intruding aerial robots as it affords a high level of visibility. As for intruding ground robots, their role on the intruding team is oriented toward detection. They choose to occupy areas where they are more likely to detect patrolling aerial robots.

Given that we can compute interpretable and reasonable patrolling behavior in this environment, we see that our method is applicable to a realistic environment, and therefore that our approach is suitable for computing patrolling strategies in real-world scenarios. Furthermore, we have shown that our game-playing model is easily extensible: We can easily incorporate sophisticated models of visibility and terrain analysis to improve the realism of the generated strategies.

9. Conclusion

We demonstrate the use of polymatrix games for effective adversarial patrolling in large-scale environments using heterogeneous agents. The proposed method is more effective in capturing intruders than independent minimax formulation but is also more amenable to decentralized implementation as compared to methods that optimize in the joint strategy space. In future work, we will implement this method on a team of decentralized robots that is robust to network failures.

Acknowledgement

We gratefully acknowledge the support of this research from the Army Research Lab - Collaborative Research Alliance - "Distributed and Collaborative Intelligent Systems and Technology" and UC San Diego Contextual Robotics Institute.

References

[1] M. S. Couceiro, D. Portugal, J. F. Ferreira, and R. P. Rocha, “Semfire: Towards a new generation of forestry maintenance multi-robot systems,” in *2019 IEEE/SICE International Symposium on System Integration (SII)*, pp. 270–276, IEEE, 2019.

[2] A. Marino, L. E. Parker, G. Antonelli, and F. Caccavale, “A decentralized architecture for multi-robot systems based on the null-space-behavioral control with application to multi-robot border patrolling,” *Journal of Intelligent & Robotic Systems*, vol. 71, no. 3, pp. 423–444, 2013.

[3] G. Brown, M. Carlyle, J. Salmerón, and K. Wood, “Defending critical infrastructure,” *Interfaces*, vol. 36, no. 6, pp. 530–544, 2006.

[4] B. M. Zoss, D. Mateo, Y. K. Kuan, G. Tokić, M. Chamanbaz, L. Goh, F. Vallegra, R. Bouffanais, and D. K. Yue, “Distributed system of autonomous buoys for scalable deployment and monitoring of large waterbodies,” *Autonomous Robots*, vol. 42, no. 8, pp. 1669–1689, 2018.

[5] D. Portugal and R. Rocha, “A survey on multi-robot patrolling algorithms,” in *Doctoral conference on computing, electrical and industrial systems*, pp. 139–146, Springer, 2011.

[6] L. Huang, M. Zhou, K. Hao, and E. Hou, “A survey of multi-robot regular and adversarial patrolling,” *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 4, pp. 894–903, 2019.

[7] D. Shishika and V. Kumar, “A review of multi agent perimeter defense games,” in *International Conference on Decision and Game Theory for Security*, pp. 472–485, Springer, 2020.

[8] S. Talebi, M. A. Simaan, and Z. Qu, “Cooperative, non-cooperative and greedy pursuers strategies in multi-player pursuit-evasion games,” in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 2049–2056, IEEE, 2017.

[9] R. Vidal, O. Shakernia, H. J. Kim, D. H. Shim, and S. Sastry, “Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation,” *IEEE transactions on robotics and automation*, vol. 18, no. 5, pp. 662–669, 2002.

[10] J. Frost and L. Stone, “Review of search theory: Advances and applications to search and rescue decision support. us coast guard research and development center rep,” tech. rep., CG-D-15-01, 2001.

[11] F. M. Delle Fave, A. X. Jiang, Z. Yin, C. Zhang, M. Tambe, S. Kraus, and J. P. Sullivan, “Game-theoretic patrolling with dynamic execution uncertainty and a case study on a real transit system,” *Journal of Artificial Intelligence Research*, vol. 50, pp. 321–367, 2014.

[12] Y. Cai, O. Candogan, C. Daskalakis, and C. Papadimitriou, “Zero-sum polymatrix games: A generalization of minmax,” *Math. Oper. Res.*, vol. 41, pp. 648–655, May 2016.

[13] P. Paruchuri, J. P. Pearce, M. Tambe, F. Ordonez, and S. Kraus, “An efficient heuristic approach for security against multiple adversaries,” in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pp. 1–8, 2007.

[14] P. Paruchuri, J. P. Pearce, J. Marecki, M. Tambe, F. Ordonez, and S. Kraus, “Playing games for security: An efficient exact algorithm for solving bayesian stackelberg games,” in *AAMAS*, 2008.

[15] Y. Vorobeychik, B. An, M. Tambe, and S. Singh, “Computing solutions in infinite-horizon discounted adversarial patrolling games,” in *Proceedings of the International Conference on Automated Planning and Scheduling*, vol. 24, 2014.

[16] T. Alam, M. Edwards, L. Bobadilla, and D. Shell, “Distributed multi-robot area patrolling in adversarial environments,” in *International Workshop on Robotic Sensor Networks*, 2015.

[17] A. X. Jiang, Z. Yin, C. Zhang, M. Tambe, and S. Kraus, “Game-theoretic randomization for security patrolling with dynamic execution uncertainty,” in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pp. 207–214, 2013.

[18] Y. Cai and C. Daskalakis, “On minmax theorems for multiplayer games,” in *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 217–234, Philadelphia, PA: Society for Industrial and Applied Mathematics, Jan. 2011.