A Survey of Advancements in Green Security Games

Joe McCall

CAP5600

Abstract

The field of Green Security Games (GSG) has proven useful in the protection of wildlife. By modelling attackers and defenders as intelligent agents in a repeated simulation we can employ a winning algorithm to deploy scarce resources in actual green security scenarios. Such an abstraction also serves as the foundation upon which more intricate scenarios can be built, explored, and subsequently adapted to by a learning agent. This paper summarizes the concept of GSGs, surveys the advancements that have been made, provides analysis of the validity of the advancements, and suggests future opportunities for research.

1 Introduction

The domain of green security entails the struggle between poachers/illegal fishers and rangers charged to protect the wildlife. The rangers are limited in number and have strict constraints applied.

In the field of Artificial Intelligence, Stackelberg Security Games (SSG) have been used to predict potential attacker moves in order to assist in defender strategies. The application of this strategy to green security domains is called Green Security Games (GSG, introduced by [1]). The game abstracts the reality of poachers versus rangers patrolling a vast wildlife area into a grid-based simulation between attackers and defenders, both of which are AI agents. The attacker's goal is to place snares for wildlife without being caught, while the defender's goal is to detect both the snare and the attacker. Furthermore, the attacker behaves human-like and does not always behave optimally.

The game is run in episodes [1]. Each round the attacker chooses a target based on the highest estimated utility, and the defender must deploy its resources judiciously to defend those targets. The defender does not have enough resources to defend all targets, so it must employ a strategy to decide which targets should be defended and which targets should be left defenseless. By simulating multiple rounds of this game, an optimized defender strategy is formed. This strategy is applicable to assist real-world rangers in deciding where and how patrols should be deployed.

Advancements in this field introduce additional constraints, challenge assumptions, and propose novel methods for improving defender strategy. Qian et al. [2] show that the environment must be partially-observable to model the defender's decision to explore new areas or patrol areas known to have snares. Wang et al. [3] introduce the idea that both attackers and defenders have access to real-time information that can be used to evade and track. Finally Gholami et al. [4] show that a defender strategy that combines a machine-learning agent with an online learning algorithm that does not rely on prior information can outperform existing models.

The paper is organized in three sections. The Methods/Theory section summarizes the methods for the these papers. The Discussions section summarizes the results, and reflects on the soundness of each paper. The Conclusions section analyzes the research, and provides ideas for further research.

2 Methods/Theory

- Include evaluation on soundness of each method presented. Why is this valid?

2.1 Green Security Games (GSG)

The Green Security Game as introduced by Fang et al. [1] is a zero-sum game. It is run in T ($< \infty$) rounds, and each round has multiple episodes. At the end of each episode the defender can change her deployment strategy. The defender has K guards to protect N ($\geq K$) targets, each with a different reward. A guard (defender) can defend one target i and an attacker can attack one target. If the attacker attacks an unguarded target, the defender is penalized and the attacker is rewarded. If the attacker attacks a defended target, the attacker is penalized and the defender is rewarded. After each round the defender assigns guards in order to maximize the expected utility.

Defender

The defender strategy is represented by a coverage vector $c = \langle c_i \rangle$ where $0 \leq c_i \leq 1$ is the probability that target i is covered by a guard. It also satisfies $\sum_{i=1}^{N} c_i \leq K$ [1]. In other words it is impossible to have more total coverage than there are guards.

To compute the utility of the defender we use fig. 1. The expected utility for a defender with a given strategy c is the probability target i is covered times the reward of that target for the defender, plus the probability that target is not covered times the penalty of that target for the defender.

$$U_i^d(c) = c_i R_i^d + (1 - c_i) P_i^d \tag{1}$$

The defender's strategy in round t is denoted by c^t.

Several algorithms are introduced to attempt to optimize the defender strategy. They include PlanAhead-M (PA-M), FixedSequence-M (FS-M), and an enhanced PA-M that incorporates Bayesian Updating (BU).

Attacker

The strategy utilized by the attacker depends on the Subject Utility Quantal Response (SUQR) concept, which has proven to accurately model bounded rationality of human attackers [5].

In other words, the attacker uses the his belief in the defender's strategy c_t and his limited memory of previous rounds to decide on the target to attack in that round.

2.2 Exploration/Exploitation Tradeoffs

Qian et al. [2] suggest that the GSGs as proposed by Fang et al. [1] fail to address an important reality in patrolling an area: the attacks on unguarded targets can only be discovered if the guards explore that area first. In other words, the environment is only partially observable by the defender. The defender must choose between patrolling targets with known poaching activities and exploring the targets that may or may not have been attacked.

Restless Multi-armed Bandit (RMAB) Problem

To solve this problem the game is modelled as a Restless Multi-armed Bandit (RMAP) problem. In such a problem, limited resources (guards) must protect several targets, but they have no insight into targets which they do not protect. It is considered "restless" because the non-activated arms transition state as well as activated arms.

Expectation-Maximization (EM) Learning Algorithm

Explain the EM learning algorithm at a high-level and how it addresses RMAB.

Whittle Index

Explain how Whittle Index is used in conjunction with the EM algorithm to help the defender plan. The Whittle index is the heuristic index policy that assists the agent in deciding which arm to activate.

2.3 Real-Time Information

- Explain what Want et al. [3] show as shortcomings of prior work, and how GSG-I augments the existing game and DeDOL helps to solve it.

GSG-I Problem

- GSG-I is an augmentation of GSG that includes real-time information
- Explain how it is closer to reality than GSG

DeDOL Algorithm

- Explain what the Double-Oracle framework is
- Expand upon the domain-specific heuristics it uses

2.4 Imperfect Prior Knowledge

- Explain what Gholami et al. [4] show as shortcomings of prior work, and how MINION-sm and MINION help improve defender strategies in GSGs.
- Historical information is unreliable because it exhibits spatial bias [4]. In other words, since we only know about attacks that we can observe, and the area around guard posts is more observed than other areas, historical data will reward guarding targets closer to the outposts much higher than other, potentially more valuable targets.
- Show how Gholami et al. [4] adjusted the GSG presented in [1] to illustrate their point.
- Introduce the two experts used in the MINION algorithm - MINION-sm and Machine Learning

MINION-sm

- Explain MINION-sm as the online-learning algorithm without historical data
- MINION-sm starts with the FPL-UE (follow-the perturbed-leader with uniform exploration) algorithm. It randomly flips a coin to choose between exploration/exploitation. It is impractical for deployments in GSGs. MINION-sm adds scheduling constraints to this algorithm. Randomness is added to ensure the route chosen isn't fixed.

Machine Learning (ML)

- Show how MINION adds a Machine-Learning element to take advantage of historical data.

MINION

- Show how patrol planning balances between the two experts
- State that the experiments showed that MINION outperformed MINION-sm and ML individually

3 Discussions

Include:

- GSG abstraction - benefits and trade-offs. What benefit does the GSG abstraction have over an attempted perfect simulation?
- RMAB - How does this improve on the GSG abstraction? Are there any assumptions being made?

- GSG-I - Same question
- MINION - Is this seemingly the ultimate solution? How would it perform against GSG-I? How can its work be combined with previous work discussed in this paper?
- Additional insight - how this problem is still not fully addressed, and what could be done in further research.

The introduction of GSG has effectively abstracted the conflict between poachers and the law enforcement agents attempting to stop them. The ability to simulate multiple rounds against a simulated human presents a clear benefit for learning algorithms to assist in the deployment of guards to patrol vast areas. Fang et al. [1] showed that the enhanced PA-M strategy provided a high average expected utility compared to the baseline algorithm used. Assuming the abstraction is valid, this indicates the strategy would be successful if employed by actual law enforcement.

However, the assumption that adversaries act with bounded rationality may only be useful in the short-term. Access to computing resources (and even the PAWS application) is growing more ubiquitous, and it's only a matter of time before poachers have the same access to the AI research as law enforcement. A means to detect whether or not the attacker is utilizing AI assistance would be highly valuable in this case.

4 Conclusions

- Summarize research
- Discuss how PAWS is helping law enforcement currently, and how these algorithms can be used to improve it
- Find a unique idea as a future research goal

5 References

[1] F. Fang, P. Stone, and M. Tambe, "When security games go green: Designing defender strategies to prevent poaching and illegal fishing," in *Proceedings of the 24th international conference on artificial intelligence*, 2015, pp. 2589–2595.

[2] Y. Qian, C. Zhang, B. Krishnamachari, and M. Tambe, "Restless poachers: Handling exploration-exploitation tradeoffs in security domains," in *Proceedings of the 2016 international conference on autonomous agents & multiagent systems*, 2016, pp. 123–131.

[3] Y. Wang *et al.*, "Deep reinforcement learning for green security games with real-time information," in *Proceedings of the thirty-third aaai conference on artificial intelligence*, 2019, pp. 1401–1408.

[4] S. Gholami, A. Yadav, L. Tran-Thanh, B. Dilkina, and M. Tambe, "Don't put all your strategies in one basket: Playing green security games with imperfect prior knowledge," in *Proceedings of the 18th international conference on autonomous agents and multiagent systems*, 2019, pp. 395–403.

[5] T. H. Nguyen, R. Yang, A. Azaria, S. Kraus, and M. Tambe, "Analyzing the effectiveness of adversary modeling in security games," in *Proceedings of the twenty-seventh aaai conference on artificial intelligence*, 2013, pp. 718–724.