AAAI Technical Report SS-12-03
Game Theory for Security, Sustainability and Health

Strategy Representation Analysis for Patrolling Games

Branislav Bošanský, Ondřej Vaněk, Michal Pěchouček
Agent Technology Center
Faculty of Electrical Engineering
Czech Technical University in Prague
{bosansky, vanek, pechoucek}@agents.felk.cvut.cz

Abstract

This paper considers the problem of patrolling multiple targets in a Euclidean environment by a single patrolling unit. We use game-theoretic approach and model the problem as a two-player zero-sum game in the extensive form. Based on the existing work in the domain of patrolling we propose a novel mathematical non-linear program for finding strategies in a discretized problem, in which we introduce a general concept of internal states of the patroller. We experimentally evaluate game value for the patroller for various graphs and strategy representations. The results suggest that adding internal states for the patroller yields better results in comparison to adding choice nodes in the used discretization.

Introduction

Game theory provides theoretic and algorithmic foundations for the field of multi-agent systems by formally modeling situations with non-cooperative self-interested autonomous agents, and defining the optimal behavior of the agents in such situations. Lately, game-theoretic methods have been particularly successful in finding solutions for security and defense scenarios that include computer networks with legitimate users and attackers, public transport system providing services to paying customers and fare-evaders, or decentralized surveillance system protecting buildings or utility infrastructure. Moreover, the game-theoretic results were successfully applied in a variety of security (Pita et al. 2008; Tsai et al. 2009; Pita et al. 2011) and transportation domains (Jakob, Vanek, and Pechoucek 2011).

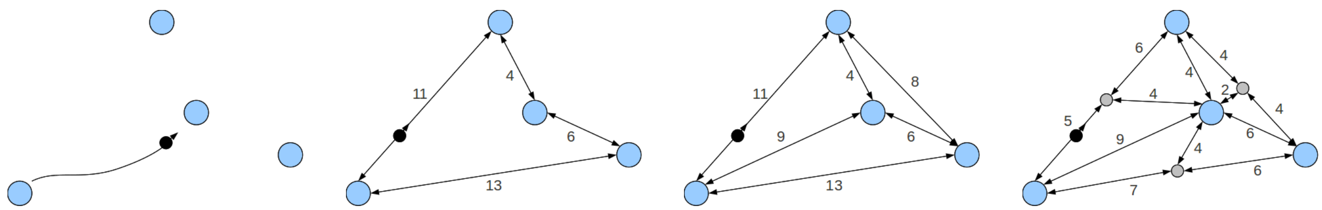
Specific variants of the security games are termed *patrolling security games* (PSGs) and constitute the problem of protecting an area, or targets placed in an environment, against an attack. The solution of a PSG is a description of a movement of a patrolling unit(s) in the area in such a way that it minimizes the probability that none of the targets will be left unvisited for longer than some given period of time. This problem can be modeled as a two-player zero-sum game in the extensive form between *the defender* (or *the patroller*), controlling possibly multiple patrolling units, and *the attacker* that aims to attack some of the targets or enter the protected area. The game evolves in turns; each turn the defender moves the patrolling units according to their capabilities, and the attacker can either choose to attack some target (the attack takes a given period of time and cannot be suspended by the attacker), or it can wait. The goal of the patroller is to discover the attacker during the attack; attacker aims for successfully finishing the attack.

Our work contributes to the broader class of patrolling games that occur in a Euclidean environment (possibly with polygonal obstacles), there is a number of targets that need to be protected by a single patrolling unit, and the targets can move in the environment in time. There is a number of real-world scenarios that are modeled by such variant of PSGs — a typical example from the maritime domain concerns a protection of vessels transiting waters with high pirate activity, another example concerns unmanned-aerial-vehicle-based surveillance protecting ground targets.

In this paper we present preliminary results of our approach. We focus on the problem of environment discretization and strategy representation and study how the properties of the graph, on which the patroller moves, affect the performance of the patroller. In this work we assume a Euclidean environment without obstacles, a single patroller, stationary targets, and zero-sum formulation of the game. We provide a novel mathematical framework that models the problem of patrolling on an arbitrary weighted graph, and introduces a general concept of states of the patroller, that, for example, correspond to the current direction of the patrolling unit in the environment. We use our mathematical formulation and evaluate the performance of existing approaches described in works by Agmon et al., or Basilico et al. for solving the patrolling problem in Euclidean space, and compare them with a model combining their characteristics. The results show that the novel model improves the existing approaches.

Related Work

In the security and defense domains of our concern, the aim of an intelligent adversary is to harm one or more targets in the environment. The defender protects the targets using limited resources by following a randomized strategy. We identify three main classes of related problems: (1) static allocation of resources to targets or nodes (e.g., allocating security guards to checkpoints at airport); (2) graph-based resource allocation problems, where the strategies of involved players depend on an underlying graphical structure of the



(a) Euclidean space (b) Perimeter-like graph (c) Directly connected targets (d) Directly connected targets with additional choice nodes

Figure 1: Visualization of the problem of patrolling in Euclidean space, the patroller is showed as a black circle in the middle with the arrow representing its direction, large blue circles represent targets; 1b-1d show three possible variants of discretization of the problem; the numbers represent the weights of the edges measured in number of steps of the patroller.

problem (e.g., placing security checkpoints on a road network); and (3) mobile patrolling in an area and periodic visiting the targets. All of the security games models assume observability of the defender's strategy. We follow the third class and focus on solving the problem of patrolling modeled by *patrolling security games* (PSGs). The main difference between PSGs in comparison to more studied resource-allocation security games (Kiekintveld et al. 2009; Yin et al. 2010) is in the extensive form of PSGs – the attacker not only knows the strategy of the defender, but can also observe its current position.

Existing approaches in patrolling security games are mostly covered by works of Agmon et al., and Basilico et al. In (Agmon, Kraus, and Kaminka 2008) the problem of perimeter patrolling is analyzed. The protected environment is discretized as a circle graph, where each node is a potential target. The authors seek the defender's strategy both as a simple Markovian policy (bidirectional movement), as well as a policy with an additional state (directional movement either with or without a cost of changing the direction). The work was lately improved by concerning complex environmental conditions in (Agmon, Urieli, and Stone 2011).

On the other hand, in (Basilico, Gatti, and Amigoni 2009) the authors extend the perimeter patrolling to patrolling targets in arbitrary graphs, computing the strategy as a Markovian policy. The work was lately extended by finding abstractions in the graph structure in (Basilico and Gatti 2011).

This work extends existing works by exploring one of the important aspects of the patrolling games, the impact of environment discretization and strategy representation on the performance of the defender, bridging the gap between target-based patrolling on arbitrary graphs and perimeter patrolling.

Problem Definition and Method

A *patrolling game* is a two-player game between *the defender* (also termed *the patroller*) and *the attacker*. The game is played in discrete time in a Euclidean environment without obstacles, where a set of targets Q is placed. We seek the strategy for the patroller in a form of randomized routes through the environment. The strategy of the attacker is a choice which target to attack and in what situation. We assume that the attacker can observe the patroller, can select only one target to attack, and can be captured while attacking the target, because the attack takes some time period to complete (we denote this time as $\mathcal{D}$ turns of the game). We use a zero-sum formulation of the game; hence, the patroller aims to minimize the probability, that some target will remain unvisited for more than $\mathcal{D}$ turns under the assumption of observability by the attacker. An example of such situation is depicted in Figure 1a. Finding an appropriate representation of the strategies of the patroller and environmental discretization is essential in solving such patrolling game.

Let us assume we have a discretization of the environment – i.e., the patroller is moving on some graph $G = (N, E)$. We assume that every target is placed in some node; hence $Q \subseteq N$. Since we do not pose any other assumptions on the topology of the graph, we define a set of weights W — $w_{(n_1,n_2)} \in W$ determines number of turns of the game (steps of the patroller) it takes for the patroller to reach a node n_2 from a node n_1. We assume that the weights are positive, smaller than $\mathcal{D}$, and they satisfy triangle inequalities.

Besides the positions, we define a set of states S, in which the patroller can be during the game. These states can represent observable characteristics (e.g., current orientation of the patrolling robot) or internal beliefs of the patroller. Not each transition from each node n_1 and state s_1 to node n_2 and state s_2 can be meaningful in the scenario; let us use the example from the perimeter patrolling – if the patroller is facing one direction and it decides to move forward, it is not possible that it will reach the following node facing the opposite direction. Thus, we define $\mathcal{T}$ as a set, where $t_{(n_1,s_1),(n_2,s_2)} \in \mathcal{T}$ equals 1 if and only if such transition is between some nodes n_1 and n_2, and states s_1 and s_2 is allowed; 0 otherwise.

Now, we can define variables $\alpha^h_{(n_1,s_1),(n_2,s_2)}$ to be a probability that the patroller will reach node n_2 and internal state s_2 starting in node n_1 and internal state s_1 in *exactly* h turns. Finally, we define an auxiliary binary variable $x_{(n_1,s_1),(n_2,s_2)}$ that indicates whether the patroller in its strategy ever uses the transition $(n_1, s_1) \to (n_2, s_2)$. Therefore, the optimal strategy α, given the graph G, states S, and transitions $\mathcal{T}$, can be found by solving the following mixed-integer non-linear program (MINLP)[1]:

$$\max_{\alpha} V \qquad s.t.$$

[1]We omit the universal quantifiers for n_1, n_2, s_1, s_2, q, h in the equations if there are no specific restrictions.

$$x_{(n_1,s_1),(n_2,s_2)} \in \{0,1\} \tag{1}$$

$$0 \le \alpha^h_{(n_1,s_1),(n_2,s_2)} \le 1 \tag{2}$$

$$\alpha^h_{(n_1,s_1),(n_2,s_2)} = 0 \qquad \forall h \in \{0,\dots,w_{(n_1,n_2)}-1\} \tag{3}$$

$$0 \le \alpha^{w_{(n_1,n_2)}}_{(n_1,s_1),(n_2,s_2)} \le x_{(n_1,s_1),(n_2,s_2)} \le M \cdot \alpha^{w_{(n_1,n_2)}}_{(n_1,s_1),(n_2,s_2)} \tag{4}$$

$$\alpha^{w_{(n_1,n_2)}}_{(n_1,s_1),(n_2,s_2)} \le t_{(n_1,s_1),(n_2,s_2)} \tag{5}$$

$$\sum_{n_i \in N} \sum_{s_i \in S} \alpha^{w_{(n_1,n_i)}}_{(n_1,s_1),(n_i,s_i)} = 1 \tag{6}$$

$$\alpha^h_{(n_1,s_1),(n_2,s_2)} = \sum_{n_i \in N\setminus\{n_2\}} \sum_{s_i \in S} \alpha^{w_{(n_1,n_i)}}_{(n_1,s_1),(n_i,s_i)} \cdot \alpha^{\max(0,h-w_{(n_1,n_i)})}_{(n_i,s_i),(n_2,s_2)} \qquad \forall h \in \{w_{(n_1,n_2)}+1,\dots,\mathcal{D}\} \tag{7}$$

$$\sum_{h=0}^{\mathcal{D}} \sum_{s_q \in S} \alpha^h_{(n_1,s_1),(q,s_q)} \ge V \tag{8}$$

$$\sum_{h=0}^{\mathcal{D}-w_{(n_k,n_1)}+1} \sum_{s_q \in S} \alpha^h_{(n_1,s_1),(q,s_q)} \ge V \cdot x_{(n_k,s_k),(n_1,s_1)} \qquad \forall n_k \in N, \forall s_k \in S : t_{(n_k,s_k),(n_1,s_1)} > 0 \tag{9}$$

The meaning of the constraints is as follows: constraints (1-2) ensure boundaries for variables α and limit the auxiliary variables x to binary values; constraints (3) ensure zero probability for the transition with insufficient time; constraints (4) ensure boundaries for probabilities of using the transition from $(n_1, s_1) \rightarrow (n_2, s_2)$ (M is some very large constant); constraints (5) ensure zero probability for not allowed transitions; constraints (6) ensure the summation of the outgoing probabilities to 1; constraints (7) represent the recursive definition of probability for higher number of steps h — the constraints say that the probability of reaching a node n_2 and a state s_2 from a (n_1, s_1) in exactly h steps can be expressed as a product of probability of transition to some other node n_i (different from node n_2) and state s_i, and probability of reaching (n_2, s_2) from (n_i, s_i) in exactly $(\mathcal{D} - w(n_1, n_i))$ steps.

Constraints (8) represent the first part of optimization constraints – we are maximizing minimal value of V that represent the patroller's worst-case probability of reaching some target in at most $\mathcal{D}$ steps. Thus, we sum the probabilities for different number of steps h (note, that α variables represent probabilities of reaching specific node in *exactly* h steps). Moreover, since in typical patrolling scenario it is irrelevant in which state we reach the target node q, we sum through all possible states. These constraints, however, are not enough for correct functioning of the model. We assume that the edges can be longer than 1 (in the number of patroller's steps), therefore we need to give the attacker an opportunity to start the attack while the patroller is on an edge. Let's assume that the patroller is situated in some node n_k and decides to go over an edge to node n_1. After one step, the patroller is situated somewhere on the edge, and now the attacker knows the deterministic course of the game for next $(w_{(n_k,n_1)} - 1)$ steps. Thus, the attacker can choose to attack some target q while the patroller is on the edge $(n_k, s_k) \rightarrow (n_1, s_1)$. Of all possible places of the patroller on this edge, starting the attack after one step is a dominating strategy for the attacker. Therefore, we need to consider the probabilities of reaching the target q from node n_1 in at most $\mathcal{D} - (w_{(n_k,n_1)} - 1) = \mathcal{D} - w_{(n_k,n_1)} + 1$ steps in optimization constraints (9) as well. Moreover, we need to weight the impact of this constraint by the variable x that indicates whether the patroller actually uses the transition $(n_k, s_k) \rightarrow (n_1, s_1)$.

The MINLP formulation can be approximated as a nonlinear program (NLP) by making the x variables continuous and modifying the constraints (4) as follows (ϵ denotes some very small constant, M denotes some large constant):

$$0 \le x_{(n_1,s_1),(n_2,s_2)} \le 1 \tag{10}$$

$$\frac{\alpha^{w_{(n_1,n_2)}}_{(n_1,s_1),(n_2,s_2)}}{\epsilon + \alpha^{w_{(n_1,n_2)}}_{(n_1,s_1),(n_2,s_2)}} \le x_{(n_1,s_1),(n_2,s_2)} \le M \cdot \alpha^{w_{(n_1,n_2)}}_{(n_1,s_1),(n_2,s_2)} \tag{11}$$

Using multiple states benefits the patroller, since it increases the expressiveness of the representation of patroller's strategies. The states can differ in observability from the attacker's perspective and the presented NLP can be modified to reflect unobservable states. We introduce variables $y(n_1, s_1)$ defined by constraints (12), which represent the probability that the patroller is in a node n_1 and a state s_1 based on the transition probabilities α. Now, constraints (8) can be replaced with constraints (13), where we sum the probability of reaching the target q over possible starting states s_i weighted by the probability of the patroller being in the state s_i if it is located in node n_1. Similarly, the constraints (9) are replaced with constraints (14), where the summation is weighted by the probability that the patroller will reach the node n_1 in state s_i if it is using the edge $n_k \rightarrow n_1$ (the x variables do not depend on the states here):

$$\sum_{n_i \in N, s_i \in S} y_{(n_i,s_i)} = 1 \qquad \sum_{n_i \in N, s_i \in S} \alpha^{w_{(n_i,n_1)}}_{(n_i,s_i),(n_1,s_1)} \cdot y_{(n_i,s_i)} = y_{(n_1,s_1)} \tag{12}$$

$$\sum_{s_i \in S} \frac{y_{(n_1,s_i)}}{\sum_{s'_i \in S} y_{(n_1,s'_i)}} \cdot \sum_{h=0}^{\mathcal{D}} \sum_{s_q \in S} \alpha^h_{(n_1,s_i),(q,s_q)} \ge V \tag{13}$$

$$\sum_{s_k \in S} \frac{y_{(n_k,s_k)}}{\sum_{s'_k \in S} y_{(n_k,s'_k)}} \cdot \sum_{s_i \in S} \frac{\alpha^{w_{(n_k,n_1)}}_{(n_k,s_k),(n_1,s_i)}}{\sum_{s'_i \in S} \alpha^{w_{(n_k,n_1)}}_{(n_k,s_k),(n_1,s'_i)}} \cdot \sum_{h=0}^{\mathcal{D}-w_{(n_k,n_1)}+1} \sum_{s_q \in S} \alpha^h_{(n_1,s_i),(q,s_q)} \ge V \cdot x_{(n_k,\bullet),(n_1,\bullet)} \qquad \forall n_k \in N, \exists s_k, s_i \in S : t_{(n_k,s_k),(n_1,s_i)} > 0 \tag{14}$$

Experimental Evaluation

We compared several different graphs and strategy representations for the problem of patrolling of 4 targets depicted in Figure 1a. More specifically, for fixed position of the targets we measured dependence of the value of the game V computed by the mathematical program defined in the previous section on varying graphs, on varying number of internal states and allowed transitions $\mathcal{T}$ between them, and on varying duration of the attack $\mathcal{D}$. We used KNITRO solver at NEOS server (Gropp and More 1997) to solve non-linear programs, however, due to the non-linearity of the program, the presented values are only locally optimal.

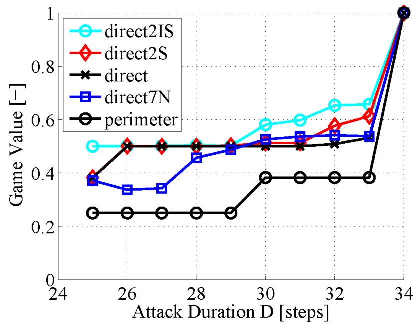


Figure 2: Results of five compared models.

We compared five representative models: Perimeter represents a directional model without a cost for turning based on works by Agmon et al., the graph has 4 nodes corresponding to the targets (as in Figure 1b), and there are two possible observable states representing the orientation of the patroller. All remaining models are based on Basilico et al.: in Direct the graph has 4 nodes, each node is directly connected with the other one (as in Figure 1c), and there is only one patroller's state; in Direct7n the graph has 7 nodes (as in Figure 1d), there is only one patroller's state; in Direct2s the graph has 4 nodes, there are two observable patroller's states, and there are no restrictions for the patroller in changing its state (in contrast to Perimeter); in Direct2is the states of the patroller are not observable by the attacker.

The results depicted in Figure 2 suggest that the patroller's strategy performs the worst for the Perimeter model, while it achieves the best values for the Direct2is model. Note, that modifying the graph by adding nodes can worsen the performance of the patroller (see the results for the Direct and Direct7n models). This phenomenon is caused by the strategy representation, in which the best strategy for the Direct model can be unrepresentable in the Direct7n model.

Conclusions and Future Work

In this paper we present preliminary results of the ongoing work on the problem of patrolling in a Euclidean environment. We proposed a mathematical framework that combines principles existing in the previous works on patrolling and we showed how different discretization of Euclidean space can affect the effectiveness of the patroller. Moreover, we introduce the general concept of states of the patroller, that can either be fully observable to the attacker (representing, for example, the current direction of the patroller), or the attacker can not observe them (representing varying intentions of the patroller). The results suggest that increasing the expressiveness of patroller's strategies using the states gives better results in comparison to adding new nodes.

However, the presented results are preliminary and further experiments on varying graphs needs to be carried out to obtain statistically significant results. Moreover, the work will continue in a number of directions and we will mainly aim to answer following questions: (1) Is it always better for the patroller to use shortest paths when moving from one target to another? (2) What is the optimal number of states of the patroller in terms of trade-off between performance and computational costs? (3) How do these characteristics change if we allow the targets to change their position in time?

Acknowledgements

The work is supported by the U.S. Army grant no. W911NF-11-1-0252, by the Czech Ministry of Education, Youth and Sports under Research Programme no. LH11051.

References

Agmon, N.; Kraus, S.; and Kaminka, G. A. 2008. Multi-robot perimeter patrol in adversarial settings. In *ICRA*, 2339–2345.

Agmon, N.; Urieli, D.; and Stone, P. 2011. Multiagent Patrol Generalized to Complex Environmental Conditions. In *AAAI Conf. on Artificial Intelligence*, 1090–1095.

Basilico, N., and Gatti, N. 2011. Automated Abstractions for Patrolling Security Games. In *AAAI Conf. on Artificial Intelligence*, 1096–1101.

Basilico, N.; Gatti, N.; and Amigoni, F. 2009. Leader-follower strategies for robotic patrolling in environments with arbitrary topologies. In *Proc. of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems*, 57–64.

Gropp, W., and More, J. 1997. Optimization Environments and the NEOS Server. In Buhmann, M. D., and Iserles, A., eds., *Approximation Theory and Optimization*, 167–182. Cambridge University Press.

Jakob, M.; Vanek, O.; and Pechoucek, M. 2011. Using agents to improve international maritime transport security. *IEEE Intelligent Systems* 26(1):90–96.

Kiekintveld, C.; Jain, M.; Tsai, J.; Pita, J.; Ordóñez, F.; and Tambe, M. 2009. Computing optimal randomized resource allocations for massive security games. In *Proc. of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems*, 689–696.

Pita, J.; Jain, M.; Western, C.; Portway, C.; Tambe, M.; Ordonez, F.; Kraus, S.; and Parachuri, P. 2008. Deployed ARMOR protection: The application of a game-theoretic model for security at the Los Angeles International Airport. In *Proc. of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems*.

Pita, J.; Tambe, M.; Kiekintveld, C.; Cullen, S.; and Steigerwald, E. 2011. Guards - game theoretic security allocation on a national scale. In *Proc. of the 10th Int. Conf. on Autonomous Agents and Multiagent Systems*.

Tsai, J.; Rathi, S.; Kiekintveld, C.; Ordóñez, F.; and Tambe, M. 2009. IRIS - A Tool for Strategic Security Allocation in Transportation Networks Categories and Subject Descriptors. In *Proc. of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems*, 37–44.

Yin, Z.; Korzhyk, D.; Kiekintveld, C.; Conitzer, V.; and Tambe, M. 2010. Stackelberg vs. Nash in security games: Interchangeability, equivalence, and uniqueness. In *Proc. of the 9th Int. Conf. on Autonomous Agents and Multiagent Systems*, 1139–1146.