Strategic Guard Placement for Optimal Response to Alarms in Security Games

(Extended Abstract)

Nicola Basilico
University of Milan
Milan, Italy
nicola.basilico@unimi.it

Nicola Gatti
Politecnico di Milano
Milan, Italy
ngatti@elet.polimi.it

ABSTRACT

We introduce a Security Game with a single static guard that is supported by a number of spatially imperfect alarms. We model this setting with two non–cooperative games modeling two different strategic interactions between a Defender and an Attacker. In the first one the Defender has to respond to an activated alarm given its current position (Alarm–Response Game). In the second one, the Defender has to determine the best static placement from which undertakes any alarm response (Guard–Placement Game).

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Multi–agent systems

General Terms

Algorithms, Economics

Keywords

Game Theory (cooperative and non-cooperative), Surveillance and Security

1. INTRODUCTION

Security Games with both mobile and static resources represent an interesting enrichment of traditional models [3] where homogeneous resources are customarily assumed [2]. Along this direction, recent works [4] suggested the study of graph patrolling problems where the Defender can control a single mobile resource (called patroller) and, at the same time, is supported by a number of static resources called alarms. Alarms are viewed as devices capable of detecting and signaling the presence of an Attacker in a vertex of the graph and characterized by detection error models (false positives and missed detection rates). In this paper, we introduce a novel security game where alarms are instead spatially imperfect, meaning that they can signal the location of an Attacker only within a subset of vertices. We propose the adoption of two linked non–cooperative games to derive the Defender's optimal patrolling response to a given alarm from a given location on the graph and its best overall placement on the graph, respectively.

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

2. PROBLEM FORMULATION

We share basic definitions with [1] by considering an undirected graph $G(V, E)$ where we denote $w_{ij} \in \mathbb{N}_{>0}$ the time needed to move from vertex i to vertex j and *vice versa*. We denote $T \subseteq V$ the set of targets, namely vertices where attacks can take place. Each target $t \in T$ has a value $\pi(t) \in (0, 1]$ and requires the intruder to spend $d(t)$ time steps to complete an attack in it. A *Defender* patrols the graph moving between vertices and spending the associated temporal cost w. At each visit of a vertex, the Defender checks for the presence of an *Attacker*.

We have a set of alarms, defined as $\{T_a : T_a \subseteq T, a \in \{1 \dots m\}\}$. Each element $T_a \subseteq T$ corresponds to a subset of targets for which the Defender could expect to receive a signal indicating that some malicious activity is taking place there. We denote this signal as alarm a, and we assume that, once issued, it provides the Defender with knowledge that a target $t \in T_a$ is under attack. If alarm a is activated, we will refer to T_a as the set of *alerted targets* and we will call *alerted graph* the complete graph $G_a = (V_a, E_a)$ where $V_a = \{\{v_D\} \cup T_a\}$, and E_a is the complete set of edges over V_a where w^a_{ij} is the shortest time to move between i and j in G.

The Defender placement is fixed at a vertex $v_D \in V$ from where it must respond to alarms, that is clearing the associated alerted graph by visiting each alerted target $t_a \in T_a$ before time $k + d(t_a)$, where k is the time at which alarm a was issued. The threat is modeled with a single Attacker that can strike a single target $t \in T$ at any time. In general, the Defender cannot cover all the targets associated to an activated alarm and, therefore, it needs to randomize over routes that clear only some targets. If the Defender chooses a route that does not include the attacked target, then the attack is successfully completed. See Fig. 1 for an example setting.

3. GAME MODELS

We study the above settings by introducing two problems which correspond to two non–cooperative games formulations.

Problem 1 (Alarm Response) *Given Defender location v_D and alarm a, what is the best graph-clearing strategy to respond to a?*

We cast this problem to the resolution of a two-player strategic-form game that we call *Alarm Response Game* (ARG) and that is described by the tuple (G_a, T_a, v_D, d, π). The set of Attacker's pure strategies is given by T_a, namely choosing which target to attack among those covered by alarm a. The Defender's pure strategies are the possible responses to a and are given by the set of *covering routes* over G_a starting from v_D: $R = \{r_1, r_2, \dots\}$, where covering route $r_i = \langle v_D, t^i_j \rangle_{j \in \{1 \dots |r_i|\}}$ is an ordered sequence of vertices such that the time required to go from v_D to a target t^i_j

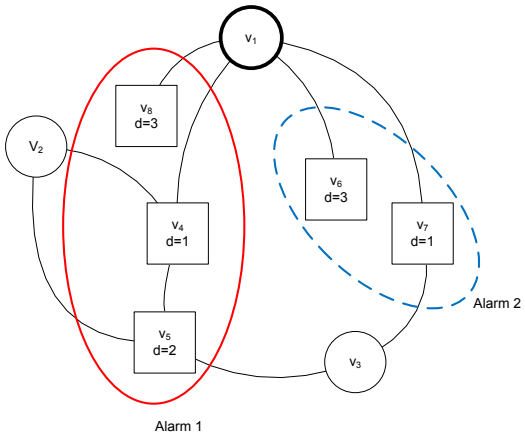

Figure 1: Example with 2 alarms. (Targets have deadlines d_t, $\pi(t) = 6$ for all t, v_1 is the optimal guard placement.)

when following sequence r_i is never larger than that target's deadline. If a route satisfies this requirement it is a *covering route*.

A strategy profile for this game is defined as $\sigma = \{\sigma_D, \sigma_A\}$ where σ_D (σ_A) is a probability distribution over the Defender's (Attacker's) pure strategies. With a slight overload of notation, we indicate a pure strategy profile as $\sigma = \{r_i, t\}$ meaning that the Defender and the Attacker assign full probability to covering route r_i and target t, respectively. Given a pure strategy profile defined in this way, payoffs are determined according to the following rule (Defender and Attacker are first and second player, respectively):

$$U(\{r_i, t\}) = \begin{cases} [1, 0] & \text{if } t \in r_i \\ [1 - \pi(t), \pi(t)] & \text{otherwise.} \end{cases}$$

The Attacker's expected utility of playing $t_a \in T_a$ is simply defined as: $EU(\sigma_D, t_a) = \sum_{r \in R_\rho} \sigma_D(r) U(\{r, t_a\})$. Being the game constant-sum, the Defender's equilibrium strategy is given by $\sigma_D^* = \arg\max_{\sigma_D} \min_{t_a \in T_a} EU(\sigma_D, t_a)$ and the game's maxmin value is denoted as $u(v_D, T_a)$.

Problem 2 (Guard Placement) *What is the best placement $v_D \in V$ to obtain maximum expected utility in responding to any alarm?*

We cast this problem to the resolution of a two-player Stackelberg game that we call *Guard Placement Game* (GPG) whose structure is: (1) the Defender selects the vertex v_D where to place itself; (2) the Attacker observes the move of the Defender; (3) the Attacker selects an alarm. Fixed v_D and an alarm a, the best strategy of the Defender (i.e., a randomization over the covering routes) and the best strategy of the Attacker (i.e., a randomization over the targets to be attacked) are uniquely determined as described by the previous game. We define this game with the tuple $(V, \{T_a\}_{a \in \{1,\ldots,m\}}, u)$, where V is the set of the actions available to the Defender, $\{T_a\}$ is the set of actions available to the Attacker, and $u : V \times \{T_a\} \to \mathbb{R}$ is the utility function of the Defender given the actions undertaken by the players. Utility function u is determined by solving the corresponding instance of an Alarm–Response Game where v_D and T_a are fixed. We can describe the game by using a matrix U in which the rows are the actions V of the Defender and the columns are the actions T_a of the Attacker, while each cell of the table is associated with utility $u(v_D, T_a)$. The Stackelberg equilibrium can be found computing $u^*(v_D) = \min_{T_a} u(v_d, T_a)$ for every v_D, and then deriving the best placement as $v_D^* = \arg\max_{v_D} u^*(v_D)$. In this way, the Defender selects the action that, once the Attacker has undertaken its action after having observed the action undertaken by the Defender, will provide the largest expected utility to the Defender.

4. DISCUSSION

Despite its simplicity, the scenario proposed in this paper poses several interesting directions of research that will be investigated in our next works towards the development of a game–theoretical framework for patrolling with alarms.

The need for patrolling. The model we introduced assumes the presence of a static Defender that selects a fixed location in the environment and takes action only when triggered by an alarm. This is an assumption that simplifies the task of computing a patrolling strategy in the broad sense in presence of alarms, i.e., a strategy that prescribes to visit different vertices at different times. As indicated by some experiments we performed with the model of [4], such task is remarkably computationally demanding. Given this drawback, the development of heuristic solutions can be worth studying in the attempt to understand how the need for patrolling changes its importance when support from alarms is available.

Spatial uncertainty. Spatially uncertain alarms represent the main novelty introduced by the above game setting. In the model presented here, such uncertainty is assumed to be spatially uniform. That is, given an activated alarm a, the Defender can derive a uniform belief over the location of the Attacker among the covered targets T_a. This assumption can be relaxed by generalizing the concept of alarm to that of *signal* which, with a certain probability, is generate by an attack in a target t. Adopting this more general setting, in principle, any probabilistic belief over the Attacker's actual location is allowed.

ARG covering routes. The most obvious computational challenge is posed by the ARG model, and concerns the computation of the Defender's actions for such a game, i.e., a set of covering routes. Given a set of targets T_a, computing the set of possible routes that guarantee to protect subsets of T_a as well as defining dominance relations among such actions is not trivial task. For this reason, after assessing the problem's complexity, algorithms to approximate such set of actions can be a useful tool to enable the resolution in realistically large settings.

Detection errors. Towards the aim of building a unified framework with [4] for patrolling with alarms, a natural extension is to include the presence of alarms which are also characterized by detection errors. The coexistence of spatial and detection uncertainties for alarms in patrolling games is a novel problem whose resolution would fit several application settings.

5. REFERENCES

[1] N. Basilico, N. Gatti, and F. Amigoni. Patrolling security games: Definition and algorithms for solving large instances with single patroller and single intruder. *ARTIF INTELL*, 184–185:78–123, 2012.

[2] N. Basilico, N. Gatti, and F. Villa. Asynchronous multi-robot patrolling against intrusion in arbitrary topologies. In *AAAI*, pages 1224–1229, 2010.

[3] M. Jain, B. An, and M. Tambe. An overview of recent application trends at the AAMAS conference: Security, sustainability, and safety. *AI Magazine*, 33(3):14–28, 2012.

[4] E. Munoz de Cote, R. Stranders, N. Basilico, N. Gatti, and N. Jennings. Introducing alarms in adversarial patrolling games. In *AAMAS*, pages 1275–1276, 2013.