Applied Soft Computing 47 (2016) 1–11

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Conforming coalitions in Markov Stackelberg security games: Setting max cooperative defenders vs. non-cooperative attackers

Julio B. Clempner[a,*], Alexander S. Poznyak[b]

[a] Centro de Investigaciones Economicas, Administrativas y Sociales, Instituto Politecnico Nacional, Lauro Aguirre 120, col. Agricultura, Del. Miguel Hidalgo, 11360 Mexico City, Mexico
[b] Department of Control Automatics, Center for Research and Advanced Studies, Av. IPN 2508, Col. San Pedro Zacatenco, 07360 Mexico City, Mexico

ARTICLE INFO

Article history:
Received 24 February 2015
Received in revised form 16 April 2016
Accepted 25 May 2016
Available online 30 May 2016

Keywords:
Security games
Stackelberg
Strong equilibrium
L_p-norm
Coalition
Multiple agents

ABSTRACT

The use of multiple agents to perform intruder search or patrolling activities have been widely studied in security Stackelberg games. Although existing methods achieve an efficient way of performing schedules, it is still an open problem that needs substantial further research in developing an algorithmic method of convergence to a Strong Stackelberg/Nash equilibrium, that would positively impact the effectiveness of the patrollers ability to execute their schedules, when acting cooperatively.

To address this shortcoming, in this work we present a novel approach for conforming coalitions in multiple agents Stackelberg security games. The concept of coalition coincides in fact with the concept of Strong Stackelberg Equilibrium. Our approach is restricted to a class of a time-discrete ergodic controllable Markov chains. The security model describes a strategic game in which the defenders cooperate and attackers do not cooperate. In case of a metric state space, the coalition of the defenders achieves it synergy by computing the Strong L_p-Stackelberg/Nash equilibrium. We employ the extraproximal method for solving the problem. In addition, we consider the control problem that involves defenders and attackers performing a discrete-time random walk. Actions are selected using the max operator, which returns the maximum value of a set of strategies. We prove that the sequences generated by the two synchronized random walks of both defenders and attackers, converge to the product of the individual probabilities. A realization of the method in a numerical example shows the efficiency of the solution.

1. Introduction

In recent years there has been extensive research on game-theoretic models in the security domain. A Stackelberg security game (SSG) is a multiple player game between a group of defenders, acting cooperatively or not, and a group of intruders (a terrorist or a criminal). The common dynamics of the game is solved as follows. Defenders consider what the best response of the attackers. The defender then commit to a mixed strategy (a probability distribution over deterministic schedules) that minimizes the costs, anticipating the predicted best response of the attackers. Then, taking into the account the adversary's mixed strategy selection, the attackers (in equilibrium) select the expected best response that maximizes the utility.

We define a Stackelberg security game as follows. Let us consider a game with $\mathcal{N}+\mathcal{M}$ players. We state a set of players as *defenders* (leaders) and let us denote by U be their strategy set. The remaining $\mathcal{M}$ players are called *attackers* (followers). Let the set of their partial strategy profiles be V, so that $U \times V$ is the set of full strategy profiles. The Stackelberg security game is the game where the defenders first choose u from the set U of all possible Stackelberg/Nash equilibria $u^* \in U$, that correspond to a strong Nash equilibrium (because in this case leaders collaborate). Then, the attackers are informed about the leaders' strategy selection (u) and they choose simultaneously their best-reply strategies in $v^* \in V$, that correspond to a Nash equilibrium (because in this case attackers do not cooperate). The cost-function $\varphi_l(\cdot, \cdot|v)$ $(l=\overline{1,\mathcal{N}})$ shows the payoffs for the defenders when they apply the strategy $u \in U$ given that the followers play strategy $v \in V$ and, the function $\varphi_m(\cdot, \cdot|u)$ $(m=\overline{1,\mathcal{M}})$ denotes the joint payoffs for the attackers given the u fixed strategy of the leaders, which is a best-reply in the original game.

Important real-world applications [1] based on SSGs have been successfully deployed: (1) ARMOR [2,3], (2) IRIS [4,5], (3) GUARDS [6,7], (4) PROTECT [6,8,1,9], and (5) RaPtoR [10]. Previous research

* Corresponding author. Tel.: +52 5548775074; fax: +52 5548775074.
E-mail addresses: julio@clempner.name (J.B. Clempner), apoznyak@ctrl.cinvestav.mx (A.S. Poznyak).

http://dx.doi.org/10.1016/j.asoc.2016.05.037
1568-4946/

has the main focus on computing the Strong Stackelberg Equilibrium (SSE), which optimizes the defender's random allocation of limited resources. For instance, Clempner and Poznyak [11] proposed computing the shortest-path to represent the Stackelberg security game as a potential game using the Lyapunov theory. Trejo et al. [12] used the extraproximal method for computing the Stackelberg/Nash equilibria in the case of one defender and a group of multiple attackers for a non-cooperative Stackelberg game. Yang et al. [13] and Nguyen et al. computed optimal strategies based on bounded rationality. An et al. [14] restrict the set of targets that attackers can successfully attack. An et al. [15] introduced a model that explicitly represents the process of an attacker observing a sequence of resource allocation decisions and updating his beliefs about the defender's strategy considering computational techniques for updating the attacker's beliefs and computing optimal strategies for both the attacker and defender, given a specific number of observations. Blum et al. [16] demonstrated using zero-sum security games that lazy defenders, who simply keep optimizing against perfectly informed attackers, are almost optimal against diligent attackers, who go to the effort of gathering a reasonable number of observations. Letchford and Vorobeychik [17] suggested a general formulation for deterministic plan interdiction as a mixed-integer program, and used constraint generation to compute optimal solutions and, in addition, presented a greedy heuristic for this problem, and compare its performance with the optimal MILP-based approach. Jain et al. [5], Yin and Tambe [18] and Jiang et al. [19] proposed to use Bayesian Stackelberg game models. Kiekintveld et al. [20] suggested an approach based on using intervals to model uncertainty in security games and presented a polynomial time algorithm. Korzhyk et al. [21] provided a framework for showing that the security game could be either a Nash or a Stackelberg game relieving the equilibrium selection problem. They also examined the case of a follower (attacker) who can attack more than one target. The authors showed that Nash and Stackelberg strategies are the same in the majority of cases only when the follower attacks just one target. They also proposed an extensive-form game model that makes the defender's uncertainty about the attacker's ability to observe explicit.

Patrolling can be conceptualized as choosing efficient ways of performing inspections to the security points of a given area [11]. It can be considered a multiagent system because the process will be started in a distributed manner, by a group of agents. Efficient patrolling scheme requires unpredictability, in the sense that the attacker will not infer when the next inspection to an specific security point will take place. In addition, if the security agency needs to execute complex patrols, cooperation can have a dramatic impact over the efficiency of the patroller's ability to carry out their schedules. In this sense, Basilico et al. [22] focused on the problem of patrolling environments with arbitrary topologies considering the smallest number of robots needed to patrol a given environment and compute the optimal patrolling strategies along several coordination dimensions. Agmon et al. [23] presented a non-deterministic patrol framework for the robots suggesting a polynomial-time algorithm for determining an optimal patrol under the Markovian strategy assumption for the robots.

The defender–attacker game-theoretic approach has been also studied in depth in many other security domains problems [22,24,25].

In this paper we present a novel approach for conforming coalitions in multiple defenders and multiple attackers Stackelberg security games. Our approach is restricted to a class of a time-discrete ergodic controllable Markov chains. In the dynamics of the game defenders cooperate and attackers do not cooperate. The coalition of the defenders is reached by computing in case of a metric state space the Strong L_p-Stackelberg–Nash equilibrium: defenders select a strategy that minimizes the distance to the utopian minimum and no other strategy produces a smaller total expected loss. This is the main solution concept for security games. Instead the attackers always try to find a Nash equilibrium. We develop the method for the Stackelberg game in terms of nonlinear programming problems implementing the Lagrange principle. In addition, we employ Tikhonov's regularization method [26,27] to ensure the convergence of the cost-functions to a unique Stackelberg/Nash equilibrium. Tikhonov regularization, named for Andrey Tikhonov, is the most commonly used method of regularization of discrete ill-posed problems. A regularization operator and a suitable value of a regularization parameter have to be chosen. This operator is usually of the form of a vector space norm. We employ the extraproximal method for solving the problem and its extension for solving Stackelberg games [28,29]. It is a two-step method such that *the first step* (the extra-proximal step) consists of a prediction which calculate the preliminary position approximation to the equilibrium point, and *the second step* is designed to find a basic adjustment of the previous prediction. The iteration process of both steps converges to a Stackelberg/Nash equilibrium point. The usefulness of the method is proved theoretically and by an application example. Our goal is to analyze a four-player Stackelberg game, two defenders and two attackers. Each equation of the extraproximal method is an optimization problem for which the minimum or maximum is solved using a quadratic programming approach. In this paper we also consider the control problem that involves defenders and attackers performing a discrete-time random walk over a finite state space. The agents actions at each time step specify the probability distribution for the next state given the current state. At each time step, defenders and attackers observe the state of the system of interest and chooses an action. Actions are selected using the *max* operator which returns the maximum value of a set of strategies. The system then transitions to its next state, with the transition probability determined by the current state and the action taken. We prove that the synchronization of the random walk of defenders and attackers converge in probability to the product of the individual probabilities.

As far as we can tell, this approach has never been reported before in the literature. These results show the potential of employing algorithmic game theory to solve real-world security allocation problems using at the same time cooperation and no cooperation approach for multiple agents, ensuring the existence of a unique equilibrium point and proposing a method for computing the convergence to an equilibrium point. As well, the method converges in exponential time to a unique Strong L_p-Stackelberg/Nash equilibrium [29].

The remainder of the paper is organized as follows. In the next section we present the necessary background needed to understand the rest of the paper. In Section 3 we suggest the Stackelberg/Nash game definitions including the L_p-Nash and Strong L_p-Stackelberg/Nash equilibrium. The extraproximal method application is developed in Section 4, including the regularized Lagrange principle and describing the proximal format. In Section 5 we present the Markov format for the extraproximal method including all the details for implementation for Markov chains games. An application example showing the effectiveness of the proposed method is presented in Section 6. We close the paper with the conclusion in Section 7.

2. Markov games

Let S be a finite set, called the *state space*, consisting of finite set of states $\{s_{(1)}, \ldots, s_{(N)}\}$, $N \in \mathbb{N}$. A *Stationary Markov chain* [30] is a sequence of S-valued random variables $s(n)$, $N \in \mathbb{N}$. The Markov chain can be represented by a complete graph whose nodes are the states, where each edge $(s_{(i)}, s_{(j)}) \in S^2$ is labeled by the transition probability. The matrix $\Pi = (\pi_{(ij)})_{(s(i),s(j))\in S} \in [0,1]^{N\times N}$ determines the evolution of the chain: for each $k \in \mathbb{N}$, the power Π^k has in each entry $(s_{(i)}, s_{(j)})$ the probability of going from state $s_{(i)}$ to state $s_{(j)}$ in exactly k steps.

Definition 1. A controllable Markov chain [31] is a 4-tuple

$$MC = \{S, A, \mathbb{K}, \Pi\} \tag{1}$$

where

- S is a finite set of states, $S \subset \mathbb{N}$, endowed with discrete a topology;
- A is the set of actions, which is a metric space. For each $s \in S$, $A(s) \subset A$ is the non-empty set of admissible actions at state $s \in S$. Without loss of generality we may take $A = \cup_{s\in S} A(s)$;
- $\mathbb{K} = \{(s,a) | s \in S, a \in A(s)\}$ is the set of admissible state-action pairs, which is a measurable subset of $S \times A$;
- $\Pi = [\pi_{(ij|k)}]$ is a stationary controlled transition matrix, where

  $$\pi_{(ij|k)} \equiv P(s(n+1) = s_{(j)} | s(n) = s_{(i)}, a(n) = a_{(k)})$$

  represents the probability associated with the transition from state $s_{(i)}$ to state $s_{(j)}$ under an action $a_{(k)} \in A(s_{(i)})$, $k = 1, \ldots, M$;

Definition 2. A Markov decision process is a pair

$$MDP = \{MC, J\} \tag{2}$$

where

- MC is a controllable Markov chain (1)
- $J : S \times \mathbb{K} \to \mathbb{R}$ is a cost function, associating to each state a real value.

The strategy (policy)

$$d_{(k|i)}(n) \equiv P(a(n) = a_{(k)} | s(n) = s_{(i)})$$

represents the probability measure associated with the occurrence of an action $a(n)$ from state $s(n) = s_{(i)}$.

The elements of the transition matrix for the controllable Markov chain can be expressed as

$$P(s(n+1) = s_{(j)} | s(n) = s_{(i)}) = \sum_{k=1}^{M} P(s(n+1) = s_{(j)} | s(n) = s_{(i)},$$
$$a(n) = a_{(k)}) d_{(k|i)}(n)$$

Let us denote the collection $\{d_{(k|i)}(n)\}$ by Δ_n as follows

$$\Delta_n = \{d_{(k|i)}(n)\}_{k=\overline{1,M},\, i=\overline{1,N}}$$

A policy $\{d_n^{loc}\}_{n\geq 0}$ is said to be *local optimal* if for each $n \geq 0$ it maximizes the conditional mathematical expectation of the utility function $J(s_{n+1})$ under the condition that the history of the process

$$F_n := \{\Delta_0, P\{s_0 = s(j)\}_{j=\overline{1,N}}; \ldots; \Delta_{n-1}, P\{s_n = s(j)\}_{j=\overline{1,N}}\}$$

is fixed and can not be changed hereafter, i.e., it realizes the "one-step ahead" conditional optimization rule

$$d_n^{loc} := arg \min_{d_n \in \Delta_n} E\{J(s_{n+1}) | F_n\} \tag{3}$$

where $J(s_{n+1})$ is the utility function at the state s_{n+1}.

The dynamic of the Stackelberg game for Markov chains is described as follows. The game consists of $\iota = \overline{1, \mathcal{M}+\mathcal{N}}$ players and begins at the initial state $s^\iota(0)$ which (as well as the states further realized by the process) is assumed to be completely measurable. Each player ι is allowed to randomize, with distribution $d^\iota_{(k|i)}(n)$, over the pure action choices $a^\iota_{(k)} \in A^\iota(s^\iota_{(i)})$, $i = \overline{1, N_\iota}$ and $k = \overline{1, M_\iota}$. The leaders correspond to $l = \overline{1, \mathcal{N}}$ and followers to $m = \overline{1, \mathcal{M}}$. At each fixed strategy of the leaders $d^l_{(k_l|i_l)}(n)$ the followers make the strategy selection $d^m_{(k_m|i_m)}(n)$ trying to realize a Nash-equilibrium. Below we will consider only stationary strategies $d^\iota_{(k|i)}(n) = d^\iota_{(k|i)}$. In the ergodic case when all Markov chains are ergodic for any stationary strategy $d^\iota_{(k|i)}$ the distributions $P^\iota(s^\iota(n+1) = s_{(j_\iota)})$ exponentially quickly converge to their limits $P^\iota(s = s_{(i)})$ satisfying

$$P^\iota(s_{(j_\iota)}) = \sum_{i_\iota=1}^{N_\iota} \left(\sum_{k_\iota=1}^{M_\iota} \pi^\iota_{(i_\iota, j_\iota | k_\iota)} d^\iota_{(k_\iota|i_\iota)} \right) P^\iota(s_{(i_\iota)}) \tag{4}$$

The cost function of each player, depending on the states and actions of all the other players, is given by the values $W^\iota_{(i_1,k_1;\ldots;i_{\mathcal{N}+\mathcal{M}},k_{\mathcal{N}+\mathcal{M}})}$, so that the "average cost function" $\mathbf{J}^\iota$ in the stationary regime can be expressed as

$$\mathbf{J}^\iota(c^1, \ldots, c^{\mathcal{M}+\mathcal{N}}) := \sum_{i_1,k_1} \cdots \sum_{i_{\mathcal{M}+\mathcal{N}},k_{\mathcal{M}+\mathcal{N}}} W^\iota_{(i_1,k_1,\ldots,i_{\mathcal{M}+\mathcal{N}},k_{\mathcal{M}+\mathcal{N}})} \prod_{\iota=1}^{\mathcal{M}+\mathcal{N}} c^\iota_{(i_\iota,k_\iota)} \tag{5}$$

where $c^\iota := [c^\iota_{(i_\iota,k_\iota)}]_{i_\iota=\overline{1,N_\iota};k_\iota=\overline{1,M_\iota}}$ is a matrix with elements

$$c^\iota_{(i_\iota,k_\iota)} = d^\iota_{(k_\iota|i_\iota)} P^\iota(s^{(\iota)} = s_{(i_\iota)}) \tag{6}$$

satisfying

$$c^{(\iota)} \in C^{(\iota)}_{adm} = \begin{cases} c^{(\iota)} : \sum_{i_\iota,k_\iota} c^\iota_{(i_\iota,k_\iota)} = 1, \quad c^\iota_{(i_\iota,k_\iota)} \geq 0, \\ \sum_{k_\iota} c^\iota_{(j_\iota,k_\iota)} = \sum_{i_\iota,k_\iota} \pi^\iota_{(i_\iota,j_\iota|k_\iota)} c^\iota_{(i_\iota,k_\iota)} \end{cases} \tag{7}$$

where

$$W^\iota_{(i_1,k_1,\ldots,i_{\mathcal{M}+\mathcal{N}},k_{\mathcal{M}+\mathcal{N}})} = \sum_{j_1} \cdots \sum_{j_{\mathcal{M}+\mathcal{N}}} J_{(i_1,j_1,k_1,\ldots,i_{\mathcal{M}+\mathcal{N}},j_{\mathcal{M}+\mathcal{N}},k_{\mathcal{M}+\mathcal{N}})^\iota(n)} \prod_{\iota=1}^{\mathcal{M}+\mathcal{N}} \pi^\iota_{(i_\iota j_\iota|k_\iota)}$$

Notice that by (6) it follows that

$$P^\iota(s_{(i_\iota)}) = \sum_{k_\iota} c^\iota_{(i_\iota,k_\iota)} \quad d^\iota_{(k_\iota|i_\iota)} = \frac{c^\iota_{(i_\iota,k_\iota)}}{\sum_{k_\iota} c^\iota_{(i_\iota,k_\iota)}} \tag{8}$$

In the ergodic case $\sum_{k_\iota} c^\iota_{(i_\iota,k_\iota)} > 0$ for all $\iota = \overline{1, \mathcal{M}+\mathcal{N}}$. The*individual aim* of each player is

$$\mathbf{J}^\iota(c^1, \ldots, c^{\mathcal{M}+\mathcal{N}}) \to \min_{c^{(\iota)} \in C^{(\iota)}_{adm}}$$

To study the existence of Pareto policies we shall first follow the well-known "scalarization" approach [32,33]. Thus, given a n-vector $\lambda > 0$ we consider the cost-function $\mathbf{J}$.

Let us introduce the variables

$$u^l := col(c^{(l)}), \quad U^l := C^{(l)}_{adm} (l = \overline{1, \mathcal{N}}), \quad U := \otimes_{l=1}^{\mathcal{N}} U^l$$

for the leaders l where col is the column operator, and let us introduce the variables

$$v^m := col\, c^{(m)}, \quad V^m := C^{(m)}_{adm} (m = \overline{1, \mathcal{M}}), \quad V := \otimes_{m=1}^{\mathcal{M}} V^m \tag{9}$$

for the m followers.

We consider a Stackelberg game where the leaders collaborate and the followers do not cooperate. Then, the definitions of the equilibria are as follows:

A **strong Nash equilibrium** for the leaders is a strategy $u^{**} = (u^{1**}, \ldots, u^{\mathcal{N}**})$ given the fixed strategy $v = (v^1, \ldots, v^{\mathcal{M}})$ of the followers such that there does not exist any $u^l \in U$, $u^l \neq u^{l^{**}}$

$$\mathbf{J}(u^{1**}, \ldots, u^l, \ldots, u^{\mathcal{N}**} | v) \leq \mathbf{J}(u^{1**}, \ldots, u^{\mathcal{N}**} | v)$$

for $u^l \in U$ and $v \in V$.

A **Nash equilibrium** for the followers is a strategy $v^* = (v^{1*}, \ldots, v^{\mathcal{M}*})$ given the strategy $u = (u^1, \ldots, u^{\mathcal{N}})$ for the followers such that

$$\mathbf{J}(v^{1*}, \ldots, v^{\mathcal{M}*}|u) \le \mathbf{J}(v^{1*}, \ldots, v^m, \ldots, v^{\mathcal{M}*}|u)$$

for any $v^m \in V$ and $u \in U$.

Remark 3. The *Pareto set* can be defined as [34,35]

$$\mathcal{P} := \left\{ x^*(\lambda) : x^*(\lambda) := arg \min_{x \in X} \left[\sum_{l}^{\mathcal{N}} \lambda_l J^l(x) \right], \quad \lambda \in S^{\mathcal{N}} \right\}$$

such that

$$S^{\mathcal{N}} := \left\{ \lambda \in \mathbb{R}^{\mathcal{N}} : \lambda \in [0, 1], \quad \sum_{l}^{\mathcal{N}} \lambda_l = 1 \right\}$$

so that

$$\mathbf{J}(x^*(\lambda)) = (J^1(x^*(\lambda)), J^2(x^*(\lambda)), \ldots, J^{\mathcal{N}}(x^*(\lambda)))$$

The *Pareto front* is defined as the image of $\mathcal{P}$ under $\mathbf{J}$ as follows

$$\mathbf{J}(\mathcal{P}) := \{(J^1(x^*(\lambda)), J^2(x^*(\lambda)), \ldots, J^{\mathcal{N}}(x^*(\lambda))) | x^* \in \mathcal{P}\}$$

The vector x^* is called a *Pareto optimal solution* for $\mathcal{P}$.

3. The Stackelberg security game

Let us consider a Stackelberg game *with* $\mathcal{N}$ *defenders* whose strategies are denoted by $u^l \in U^l (l = \overline{1, \mathcal{N}})$ where U is a convex and compact set. Denote by $u = (u^1, \ldots, u^{\mathcal{N}})^{\mathrm{T}} \in U$ the joint strategy of the players and $u^{\hat{l}}$ is a strategy of the rest of the players adjoint to u^l, namely,

$$u^{\hat{l}} := (u^1, \ldots, u^{l-1}, u^{l+1}, \ldots, u^{\mathcal{N}})^{\mathrm{T}} \in U^{\hat{m}} := \otimes_{h=1,\, h \neq l}^{\mathcal{N}} U^h$$

such that $u = (u^l, u^{\hat{l}}) (l = \overline{1, \mathcal{N}})$. As well, let us consider $\mathcal{M}$ attackers with strategies $v^m \in V^m (m = \overline{1, \mathcal{M}})$ and V is also a convex and compact set. Denote by $v = (v^1, \ldots, v^{\mathcal{M}}) \in V := \otimes_{m=1}^{\mathcal{M}} V^l$ the joint strategy of the attackers and $v^{\hat{m}}$ is a strategy of the rest of the players adjoint to v^m, namely,

$$v^{\hat{m}} := (v^1, \ldots, v^{m-1}, v^{m+1}, \ldots, v^{\mathcal{M}})^{\mathrm{T}} \in V^{\hat{m}} := \otimes_{q=1,\, q \neq m}^{\mathcal{M}} V^q$$

such that $v = (v^m, v^{\hat{m}}) (m = \overline{1, \mathcal{M}})$.

3.1. The security game

Leaders and attackers together are in a Stackelberg game [36]: the model involves two Nash games restricted by a Stackelberg game defined as follows.

Definition 4. A *game with* $\mathcal{N}$*defenders and* $\mathcal{M}$ attackers said to be **a cooperative Stackelberg/Nash** game if

$$G_{L_p}(u(\lambda), \hat{u}(u, \lambda)|v) := \sum_{l=1}^{\mathcal{N}} [|\varphi_l(\overline{u}^l, u^{\hat{l}}|v) - \varphi_l(u^l, u^{\hat{l}}|v)|^p]^{1/p}$$

for $p \ge 1$ satisfies the properties:

- for the defenders

$$\max_{\hat{u}(u) \in \hat{U}} g(u, \hat{u}(u)) = \sum_{l=1}^{\mathcal{N}} \varphi_l(\overline{u}^l, u^{\hat{l}}) - \varphi_l(u^l, u^{\hat{l}}) \le 0$$

where $u^{\hat{l}}$ is a strategy of the rest of the defenders adjoint to u^l, namely,

$$u^{\hat{l}} := (u^1, \ldots, u^{l-1}, u^{l+1}, \ldots, u^{\mathcal{N}}) \in U^{\hat{l}} := \otimes_{h=1,\, h \neq l}^{\mathcal{N}} U^h$$

and

$$\overline{u}^l := arg \min_{u^l \in U^l} \varphi_l(u^l, u^{\hat{l}}|v)$$

- for the attackers

$$F_{L_p}(v, \hat{v}(v)|u) := \sum_{m=1}^{\mathcal{M}} [|\psi_m(\overline{v}^m, v^{\hat{m}}|u) - \psi_m(v^m, v^{\hat{m}}|u)|^p]^{1/p}$$

and

$$\max_{\hat{v}(v) \in \hat{V}} f(v, \hat{v}(v)) = \sum_{m=1}^{\mathcal{M}} \psi_m(\overline{v}^m, v^{\hat{m}}) - \psi_m(v^m, v^{\hat{m}}) \le 0$$

where $v^{\hat{m}}$ is a strategy of the rest of the attackers adjoint to v^m, namely,

$$v^{\hat{m}} := (v^1, \ldots, v^{m-1}, v^{m+1}, \ldots, v^{\mathcal{M}}) \in V^{\hat{m}} := \otimes_{q=1,\, q \neq l}^{\mathcal{M}} V^q$$

and

$$\overline{v}^m := arg \min_{v^m \in V^m} \psi_m(v^m, v^{\hat{m}}|u)$$

Definition 5. Let $G_{L_p}(u, \hat{u}(u)|v)$ be the cost functions of the defenders $(l = \overline{1, \mathcal{N}})$ and $F_{L_p}(v, \hat{v}(v)|u)$ the cost functions of the attackers $(m = \overline{1, \mathcal{M}})$. A strategy $u^* \in U$ of the defenders together with the collection $v^* \in V$ of the attackers is said to be **a Stackelberg/Nash equilibrium** if

$$(u^*, v^*) \in arg \min_{u \in U, \hat{u}(u) \in \hat{U}, \lambda \in \mathcal{S}^{\mathcal{N}}} \max_{v \in V, \hat{v}(v) \in \hat{V}} \{G_{L_p}(u(\lambda), \hat{u}(u, \lambda)|v) | g(u, \hat{u}(u)|v) \le 0, f(v, \hat{v}(v)|u) \le 0\} \tag{10}$$

Remark 6. If $G_{L_p}(u(\lambda), \hat{u}(u, \lambda)|v)$ and $F_{L_p}(v, \hat{v}(v)|u)$ are strictly convex then

$$(u^*, v^*) = arg \min_{u \in U, \hat{u}(u) \in \hat{U}, \lambda \in \mathcal{S}^{\mathcal{N}}} \max_{v \in V, \hat{v}(v) \in \hat{V}} \{G_{L_p}(u(\lambda), \hat{u}(u, \lambda)|v) | g(u, \hat{u}(u)|v) \le 0, f(v, \hat{v}(v)|u) \le 0\}$$

4. The extraproximal method application

4.1. The regularized Lagrange principle

Applying the Lagrange principle for Definition 5, we may conclude that (10) can be rewritten as

$$(x^*, w^*) \in arg \min_{u \in U, \hat{u}(u) \in \hat{U}, \lambda \in \mathcal{S}^{\mathcal{N}}} \max_{v \in V, \hat{v}(v) \in \hat{V}\, \omega \ge 0, \xi \ge 0} \mathcal{L}(u, \hat{u}(u), v, \hat{v}(v), \omega, \xi)$$

where

$$\mathcal{L}(u, \hat{u}(u), v, \hat{v}(v), \omega) := G_{L_p}(u(\lambda), \hat{u}(u, \lambda)|v) + \omega g(u, \hat{u}(u)|v) + \xi f_\delta(v, \hat{v}(v)|u) \tag{11}$$

The approximative solution obtained by the Tikhonov regularization is given by

$$(x^*, w^*) \in arg \min_{u \in U, \hat{u}(u) \in \hat{U}, \lambda \in \mathcal{S}^N} \max_{v \in V, \hat{v}(v) \in \hat{V}, \omega \geq 0, \xi \geq 0} \mathcal{L}(u, \hat{u}(u), v, \hat{v}(v),$$

$$\lambda, \omega, \xi)\mathcal{L}_\delta(u, \hat{u}(u), v, \hat{v}(v), \lambda, \omega, \xi) := G_{L_p,\delta}(u(\lambda), \hat{u}(u, \lambda)|v)$$

$$+ \omega g_\delta(u, \hat{u}(u)|v) + \xi f_\delta(v, \hat{v}(v)|u) - \frac{\delta}{2}(\omega^2 + \xi^2) \tag{12}$$

where

$$G_{L_p,\delta}(u(\lambda), \hat{u}(u, \lambda)|v) = \sum_{l=1}^{\mathcal{N}} [|\varphi_l(\overline{u}^l, u^{\hat{l}}|v) - \varphi_l(u^l, u^{\hat{l}}|v)|^p]^{1/p}$$

$$+ \frac{\delta}{2}(\|u\|^2 + \|\hat{u}(u)\|^2 + \|\lambda\|^2)$$

$$g_\delta(u, \hat{u}(u)|v) = \sum_{l=1}^{\mathcal{N}} \varphi_l(\overline{u}^l, u^{\hat{l}}|v) - \varphi_l(u^l, u^{\hat{l}}|v) + \frac{\delta}{2}(\|u\|^2 + \|\hat{u}(u)\|^2$$

$$+ \|\lambda\|^2)$$

$$f_\delta(v, \hat{v}(v)|u) = \sum_{m=1}^{\mathcal{M}} [\psi_m(\overline{v}^m, v^{\hat{m}}|u) - \psi_m(v^m, v^{\hat{m}}|u)] - \frac{\delta}{2}(\|v\|^2$$

$$+ \|\hat{v}(v)\|^2)$$

Now, the function $G_\delta(u, \hat{u}(u)|v)$ is strictly convex if the Hessian matrix is positive semi-definite, then $G_\delta(u, \hat{u}(u)|v)$ attains a minimum at $(u, \hat{u}(u)|v)$ if

$$\nabla^2 G_\delta(u, \hat{u}(u)|v)$$

$$= \begin{bmatrix} \frac{\partial^2}{(\partial u_1)^2} G_\delta(u, \hat{u}(u)|v) & \dots & \frac{\partial^2}{\partial u_1 \partial u_{\mathcal{N}}} G_\delta(u, \hat{u}(u)|v) \\ \frac{\partial^2}{\partial u_2 \partial u_1} G_\delta(u, \hat{u}(u)|v) & \dots & \frac{\partial^2}{\partial u_2 \partial u_{\mathcal{N}}} G_\delta(u, \hat{u}(u)|v) \\ \dots & \dots & \dots \\ \frac{\partial^2}{\partial u_{\mathcal{N}} \partial u_1} G_\delta\left(u, \hat{u}(u)|v\right) & \dots & \frac{\partial^2}{(\partial u_{\mathcal{N}})^2} G_\delta(u, \hat{u}(u)|v) \end{bmatrix}$$

$$= \begin{bmatrix} \delta I_{n_1 \times n_1} & \mathcal{DG}_{1,2}(\hat{u}_{1,2}) & \dots & \mathcal{DG}_{1,\mathcal{N}}(\hat{u}_{1,\mathcal{N}}) \\ \mathcal{DG}_{2,1}(\hat{u}_{2,1}) & \delta I_{n_2 \times n_2} & \dots & \mathcal{DG}_{3,2}(\hat{u}_{3,2}) \\ \dots & \dots & \dots & \dots \\ \mathcal{DG}_{3,1}(\hat{u}_{3,1}) & \mathcal{DG}_{3,2}(\hat{u}_{3,2}) & \dots & \delta I_{n_{\mathcal{N}} \times n_{\mathcal{N}}} \end{bmatrix} > 0$$

or, equivalently, δ should provide the inequality

$$\min_{u \in U, \hat{u} \in \hat{U}} [\lambda_{\min}(\nabla^2 G_\delta(u, \hat{u}(u)|v))] > 0 \tag{13}$$

Here, $\hat{u}_{ik}$ is independent of $u^{(i)}$ and $u^{(k)}$, that is, $(\partial/\partial u^{(i)})\hat{u}_{ik} = 0$ and $(\partial/\partial u^{(k)})\hat{u}_{ik} = 0$.

As well as, the function $F_\delta(v, \hat{v}(v)|u)$ is strictly concave if the Hessian matrix is negative semi-definite, then $F_\delta(v, \hat{v}(v)|u)$ attains a maximum at $(v, \hat{v}(v)|u)$ if

$$\max_{v \in V, \hat{v} \in \hat{V}} [\lambda_{\max}(\nabla^2 F_\delta(v, \hat{v}(v)|u))] < 0 \tag{14}$$

It is evident that, for $\delta = 0$, the problem (12) converts to problem (11).

4.2. The extraproximal method

The general format iterative version ($n = 0, 1, \dots$) of the extraproximal method with some fixed admissible initial values ($u_0 \in U$, $\hat{u}_0 \in \hat{U}$, $v_0 \in V$, $\hat{v}_0 \in \hat{V}$, $\omega_0 \geq 0$, $\xi_0 \geq 0$ and $\lambda_0 \in \mathcal{S}^{\mathcal{N}}$) is as follows [28]

1. The *first half-step* (prediction):

$$\overline{\omega}_n = arg \min_{\omega \geq 0} \left\{ \frac{1}{2}\|\omega - \omega_n\|^2 - \gamma L_\delta(u_n, \hat{u}_n(u), v_n, \hat{v}_n(v), \lambda_n, \omega, \overline{\xi}_n) \right\}$$

$$\overline{\xi}_n = arg \min_{\xi \geq 0} \left\{ \frac{1}{2}\|\xi - \xi_n\|^2 - \gamma L_\delta(u_n, \hat{u}_n(u), v_n, \hat{v}_n(v), \lambda_n, \overline{\omega}_n, \xi) \right\}$$

$$\overline{u}_n = arg \min_{u \in U} \left\{ \frac{1}{2}\|u - u_n\|^2 + \gamma L_\delta(u, \hat{u}_n(u), v_n, \hat{v}_n(v), \lambda_n, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$\overline{\hat{u}}_n = arg \min_{\hat{u} \in \hat{U}} \left\{ \frac{1}{2}\|\hat{u} - \hat{u}_n\|^2 + \gamma L_\delta(u_n, \hat{u}(u), v_n, \hat{v}_n(v), \lambda_n, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$\overline{v}_n = arg \min_{v \in V} \left\{ \frac{1}{2}\|v - v_n\|^2 - \gamma L_\delta(u_n, \hat{u}_n(u), v, \hat{v}_n(v), \lambda_n, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$\overline{\hat{v}}_n = arg \min_{\hat{v} \in \hat{V}} \left\{ \frac{1}{2}\|\hat{v} - \hat{v}_n\|^2 - \gamma L_\delta(u_n, \hat{u}_n(u), v_n, \hat{v}(v), \lambda_n, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$\overline{\lambda}_n = arg \min_{\lambda \in \mathcal{S}^N} \left\{ \frac{1}{2}\|\lambda - \lambda_n\|^2 + \gamma L_\delta(u_n, \hat{u}_n(u), v_n, \hat{v}_n(v), \lambda, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

2. The *second* (basic) half-step

$$\omega_{n+1} = arg \min_{\omega \geq 0} \left\{ \frac{1}{2}\|\omega - \omega_n\|^2 - \gamma L_\delta(\overline{u}_n, \overline{\hat{u}}_n(u), \overline{v}_n, \overline{\hat{v}}_n(v), \overline{\lambda}_n, \omega, \overline{\xi}_n) \right\}$$

$$\xi_{n+1} = arg \min_{\xi \geq 0} \left\{ \frac{1}{2}\|\xi - \xi_n\|^2 - \gamma L_\delta(\overline{u}_n, \overline{\hat{u}}_n(u), \overline{v}_n, \overline{\hat{v}}_n(v), \overline{\lambda}_n, \overline{\omega}_n, \xi) \right\}$$

$$u_{n+1} = arg \min_{u \in U} \left\{ \frac{1}{2}\|u - u_n\|^2 + \gamma L_\delta(u, \overline{\hat{u}}_n(u), \overline{v}_n, \overline{\hat{v}}_n(v), \overline{\lambda}_n, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$\hat{u}_{n+1} = arg \min_{\hat{u} \in \hat{U}} \left\{ \frac{1}{2}\|\hat{u} - \hat{u}_n\|^2 + \gamma L_\delta(\overline{u}_n, \hat{u}(u), \overline{v}_n, \overline{\hat{v}}_n(v), \overline{\lambda}_n, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$v_{n+1} = arg \min_{v \in V} \left\{ \frac{1}{2}\|v - v_n\|^2 - \gamma L_\delta(\overline{u}_n, \overline{\hat{u}}_n(u), v, \overline{\hat{v}}_n(v), \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$\hat{v}_{n+1} = arg \min_{\hat{v} \in \hat{V}} \left\{ \frac{1}{2}\|\hat{v} - \hat{v}_n\|^2 - \gamma L_\delta(\overline{u}_n, \overline{\hat{u}}_n(u), \overline{v}_n, \hat{v}(v), \overline{\lambda}_n, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

$$\lambda_{n+1} = arg \min_{\lambda \in \mathcal{S}^{\mathcal{N}}} \left\{ \frac{1}{2}\|\lambda - \lambda_n\|^2 + \gamma L_\delta(\overline{u}_n, \overline{\hat{u}}_n(u), \overline{v}_n, \overline{\hat{v}}_n(v), \lambda, \overline{\omega}_n, \overline{\xi}_n) \right\}$$

5. Realization of the security game

5.1. Basics

Markov decision processes involve a popular framework for sequential decision-making in a random dynamic environment [31]. At each time step, an agent observes the state of the system of interest and chooses an action. In the basic MDP framework, it is assumed that the strategies are previously calculated. The system then transitions to its next state, with the transition probability determined by the current state and the action taken.

To illustrate the problem, let us consider that defenders and attackers can travel from vertex to vertex of a connected, undirected graph [11]. At each time-step n ($n = 0, 1, \ldots$) and being in a state $s(n) \in S$ a player is able to select an action $a(n) \in A$. A transition model specifies how the world changes when an action is executed. The transition matrix $\Pi = (\pi_{ij})_{(s_i,s_j)\in S} \in [0,1]^{S\times S}$ determines the evolution of the Markov chain: for each $N \in \mathbb{N}$, Π^n has in each entry (s_i, s_j) the probability of going from state s_i to state s_j in exactly n steps. The distance $\kappa(s_0, s^*)$ between two states, for example, s_0 and s^* may be defined as follows:

$$\kappa(s_0, s^*) := \sum_{t=0}\sum_{i=1}\sum_{j=1}\chi(s(t) = s_i, s(t+1) = s_j)\cdot[1 - \chi(s_j = s^*)]$$

where

$$\chi(s(t) = s_i, s(t+1) = s_j) := \begin{cases} 1 & s(t) = s_i \wedge s(t+1) = s_j \\ 0 & \text{otherwise} \end{cases}$$

and

$$\chi(s_j = s^*) := \begin{cases} 1 & s_j = s^* \\ 0 & \text{otherwise} \end{cases}$$

The value $\kappa(s_0, s^*)$, intuitively speaking, expresses the desirability of state s^*.

5.2. Patrol scheduler

We start the process at the current point by fixing the strategies matrix $d^l_{(k_l|i_l)} = c^l_{i_l,k_l}/\sum_{k_l} c^l_{i_l,k_l}$ (see Eq. (8)). The goal is to determine a stationary policy that minimizes (maximizes) the long run expected cost (utility). This is done by the realization of the extraproximal method. We seek to improve $c^l_{i_l,k_l}$ by selecting the policy with the largest probability. This is intuitive, and mathematically can be expressed defining a projector Pr as follows

$$\Pr(d^l_{(k_l|i_l)}) = d^{l*}_{(k^*_l|i_l)} = \delta_{k^*_l(i_l),i_l} \tag{15}$$

where $\delta_{k^*_l(i_l),i_l}$ is the Kronecker symbol

$$\delta_{k^*_l(i_l),i_l} = \begin{cases} 1 & k_l(i_l) = k^*_l(i_l) \\ 0 & \text{otherwise} \end{cases}$$

and $k^*_l(i_l)$ is an index for which

$$k^*_l(i_l) = \max_{k_l \in M_l}(d^l_{(k_l|i_l)}) \tag{16}$$

As well, for the attackers we have that

$$\Pr(d^m_{(k_m|i_m)}) = d^{m*}_{(k^*_m|i_m)} = \delta_{k^*_m(i_m),i_m} \tag{17}$$

where $\delta_{k^*_m(i_m),i_m}$ is the Kronecker symbol

$$\delta_{k^*_m(i_m),i_m} = \begin{cases} 1 & k_m(i_m) = k^*_m(i_m) \\ 0 & \text{otherwise} \end{cases}$$

and $k^*_m(i_m)$ is an index for which

$$k^*_m(i_m) = \max_{k_m \in M_m}(d^m_{(k_m|i_m)})$$

Let $(\Omega, \mathcal{F}, P)$ be a probability space where Ω is a sample space; $\mathcal{F}$ is a -algebra of measurable subsets (events) of Ω; and P is a probability measure on $\mathcal{F}$ (P satisfies the following Kolmogorov axioms [37]).

Let us introduce the *capture condition* at time n (defender and attacker are located at the same state) as follows:

$$\sum_{j=1}^{N}\chi(w : s^l(n) = s_j \wedge s^m(n) = s_j)$$

$$= \sum_{j=1}^{N}\chi(w : s^l(n) = s_j)\chi(w : s^m(n) = s_j)$$

where $w \in \Omega$ is a trajectory.

Now, the capture event of all the attackers is given by

$$\sum_{l=1}^{\mathcal{N}}\sum_{m=1}^{\mathcal{M}}\sum_{j=1}^{N}\chi(w : s^l(n) = s_j)\chi(w : s^m(n) = s_j) \tag{18}$$

Then, considering that

$$P\{w : A \in \mathcal{F}\} = E\{\chi(w : A \in \mathcal{F})\}$$

we have that the total probability P_n of converging to a state j at time n for all the defenders and attackers is given by

$$P_{j,n} = \sum_{l=1}^{\mathcal{N}}\sum_{m=1}^{\mathcal{M}} P^l_{j,n}\{w : s^l(n) = s_j\}P^m_{j,n}\{w : s^m(n) = s_j\} \tag{19}$$

where

$$P^l_{j,n}\{w : s^l(n) = s_{j_l}\} = \sum_{i_l=1}^{N_l}\sum_{k_l=1}^{M_l}\pi^l_{i_lj_l|k_l}d^{l*}_{(k^*_l|i_l)}P^l_{n-1}\{w : s^l(n-1) = s_i\}$$

and

$$P^m_{j,n}\{w : s^m(n) = s_{j_m}\}$$

$$= \sum_{i_m=1}^{N_m}\sum_{k_m=1}^{M_m}\pi^m_{i_mj_m|k_m}d^{m*}_{(k^*_m|i_m)}P^m_{n-1}\{w : s^m(n-1) = s_{i_m}\}$$

Now, defining

$$\Pi^{l*}_{i_lj_l} = \sum_{k_l=1}^{M_l}\pi^l_{i_lj_l|k_l}d^{l*}_{(k^*_l|i_l)} \tag{20}$$

and

$$\Pi^{m*}_{i_mj_m} = \sum_{k_m=1}^{M_m}\pi^m_{i_mj_m|k_m}d^{m*}_{(k^*_m|i_m)} \tag{21}$$

we have

$$P^l_{j,n}\{w : s^l_n = s_{j_l}\} = \sum_{i_l=1}^{N_l}\Pi^{l*}_{i_lj_l}P^l_{j,n-1}\{w : s^l(n-1) = s_{i_l}\}$$

and

$$P^m_{j,n}\{w : s^m_n = s_{j_m}\} = \sum_{i_m=1}^{N_m}\Pi^{m*}_{i_mj_m}P^m_{j,n-1}\{w : s^m(n-1) = s_{i_m}\}$$

In vector format we have that

$$\mathbf{P}_n^l := (P_{1,n}^l, \ldots, P_{N,n}^l)^{\mathrm{T}}$$

$$\mathbf{P}_n^m := (P_{1,n}^m, \ldots, P_{N,n}^m)^{\mathrm{T}}$$

Then, we have

$$\mathbf{P}_n^l = \left(\left(\mathbf{\Pi}^{l*}\right)^{\mathrm{T}}\right)^n \mathbf{P}_0^l$$

$$\mathbf{P}_n^m = \left(\left(\mathbf{\Pi}^{m*}\right)^{\mathrm{T}}\right)^n \mathbf{P}_0^m$$

Then, the probability of the components of the state-vector $P_{j,n} = (P_{1,n}, \ldots, P_{N,n})^{\mathrm{T}}$ satisfies the following relation

$$P_{j,n} = \sum_{l=1}^{\mathcal{N}} \sum_{m=1}^{\mathcal{M}} P_{j,n}^l P_{j,n}^m \tag{22}$$

Given the finite state space S and let $s_1 \in S$ be a fixed initial state. Consider the defenders performing a controlled random walk on S in response to the dynamics of the attackers. The interaction between defenders and attackers proceeds in Algorithm 7 as follows:

Scheduler: Defender–Attacker
$l = \overline{1, \mathcal{N}}$ and $m = \overline{1, \mathcal{M}}$. . ..

(1) Compute the stationary policies for defenders and attackers
(2) Compute the optimal transition matrices $\Pi_{i_l j_l}^{l*}$ Eq. (20) and $\Pi_{i_m j_m}^{m*}$ Eq. (21)
(3) (a) The defender l selects from $P_n\{w : s^l(n) = s_j\}$ the state $s(j_l)$ with random $j_l \in \{\overline{1, N_l}\}$
(b) The attacker m selects from $P_n\{w : s_n^m = s_j\}$ the state $s(j_m)$ with random $j_m \in \{\overline{1, N_m}\}$
(4) Add the states $s(j_l)$ and $s(j_m)$ to the patrol schedule and draw $s(j_l) \sim P_n\{w : s^l(n) = s_j\}$ and $s(j_m) \sim P_n\{w : s_n^m = s(j)\}$
(5) Repeat steps (3), (4) until the capture event (18) is satisfied for all the attackers

Algorithm 7 *(Patrol scheduler).* The dynamics of the patrol scheduler is as follows. For the defender l $(l = \overline{1, \mathcal{N}})$ the optimal strategy is taken deterministically from $d_{(k_l^*|i_l)}^{l*}$ which is used to compute $\mathbf{\Pi}^{l^*}$. Then, the defender l determines the next state $s(j_l)$ to be considered in the patrol schedule: at each step selects randomly $j_l \in \{\overline{1, N_l}\}$ distributed according to the stochastic vector $P_n\{w : s^l(n) = s_j\}$. If the selected transition has zero probability (i.e., $P_n\{w : s^l(n) = s_j\} = 0$), the defender chooses randomly a new $j_l \in \{\overline{1, N_l}\}$ and repeat the process. Next, the defender adds the state $s(j_l)$, found in the previous step, to the patrol schedule. The attacker m $(m = \overline{1, \mathcal{M}})$ proceeds similarly but using the stochastic vector $P_n\{w : s^m(n) = s_j\}$. That completes a single iteration. The process continue iterating until the capture event (18) is satisfied for all the attackers and, then, the patrol schedule is completed.

5.3. Convergence

Remark 8. The assumption that the Markov chains are ergodic ensures that π_{ij} has a unique everywhere positive invariant distribution P_n [31] and, for a finite S, it is equivalent to the existence of some $N \in \mathbb{N}$ such that

$$(\pi_{ij})^n > 0$$

We also suppose that the ergodicity coefficient given by

$$\gamma_{erg} := \min_{n_0} \max_{j=1,\ldots,N} \min_{i=1,\ldots,N} \pi_{ij}(n) \tag{23}$$

is strictly positive, that is $\gamma_{erg} > 0$, which guarantees that the convergence to P^* is exponentially fast [30] (so that $\pi_{ij}(n)$ is geometrically ergodic).

With these assumptions clarified, we are now ready to state the following theorem:

Theorem 9. *The component of the vector $P_{j,n}$ given in Eq. (22) converge to a final distribution $P_{j,n}^*$.*

Proof. The distribution $P_{j,n}^*$ exists because the ergodicity property of the considered Markov chains. By Eq. (22) we have that

$$|P_{j,n}^l P_{j,n}^m - P_j^{l*} P_j^{m*}| = |P_{j,n}^l P_{j,n}^m - P_{j,n}^l P^{m*} + P_{j,n}^l P_j^{m*} + P_j^{l*} P_j^{m*}|$$
$$\leq |P_{j,n}^l \left(P_{j,n}^m - P_j^{m*}\right)| + |P_j^{m*}(P_{j,n}^l - P_j^{l*})| \leq |P_{j,n}^l - P_j^{l*}| + |P_{j,n}^m - P_j^{m*}|$$

Then,

$$\lim_{n\to\infty} |P_{j,n}^l P_{j,n}^m - P_j^{l*} P_j^{m*}| \leq \lim_{n\to\infty} |P_{j,n}^l - P_j^{l*}| + |P_{j,n}^m - P_j^{m*}| = 0$$

□

Remark 10. The process continue iterating until the capture event (18) is satisfied for all the attackers and, then, the patrol schedule is completed.

Theorem 11. *The computational complexity of the patrol schedule is given by*

$$O(\mathcal{N}\mathcal{M}N)$$

Proof. From Eq. (19) considering $\mathcal{N}$ defenders, $\mathcal{M}$ attackers and N states we have that the computational complexity of the patrol schedule is given by $O(\mathcal{N}\mathcal{M}N)$. □

6. Numerical example

We analyze a randomized defense-attacker realization of the game for the norm $p = 1$ which involves four players: two defenders and two attackers. The realization is simulated in rounds and in each round both, defenders and attackers are located at a state of the Markov chain. Between rounds both the defenders and attackers can stay at the current state or move to another state. Defenders and attackers are assumed to be not restricted in the sense that they can move to any state, this is because the ergodicity condition of the Markov chains.

An attacker is captured as soon as a defender and an attacker are located at the same state in a round. The realization of the game is finished when both attackers are captured. The main objective of the defenders is to capture the attackers in as few steps as possible, while the attackers goals are to maximize the number of rounds until they are caught. All the players use the randomized strategy fixed by computing the Stackelberg/Nash equilibrium. Given the randomized defenders strategy, the escape length for that strategy is the worst case expected number of rounds it takes the defenders to capture the attackers, where the worst case is with regards to all (possibly randomized) attackers strategies.

During the realization of the game the players has complete information about the strategies of the other players. However, the players have no information about the state position of the other players. The interaction between players occurs when the game ends because defenders and attackers move to the same state.

Let $N_1 = N_2 = N_3 = N_4 = 5$ be the number of states, $M_1 = M_2 = M_3 = M_4 = 2$ be the number of actions of the defenders ($l = 1, 2$) and the followers ($m = 3, 4$), $\gamma = 0.0007$. The transition matrices are established as follows

$$\pi^{(1)}_{ij|1} = \begin{bmatrix} 0.9469 & 0.0244 & 0.0050 & 0.0081 & 0.0156 \\ 0.1781 & 0.0406 & 0.0739 & 0.0515 & 0.6558 \\ 0.1172 & 0.0531 & 0.7386 & 0.0514 & 0.0398 \\ 0.5900 & 0.0419 & 0.0917 & 0.0231 & 0.2533 \\ 0.4804 & 0.1197 & 0.3204 & 0.0373 & 0.0422 \end{bmatrix}$$

$$\pi^{(1)}_{ij|2} = \begin{bmatrix} 0.2946 & 0.0489 & 0.0070 & 0.0888 & 0.5607 \\ 0.2401 & 0.6022 & 0.0414 & 0.0745 & 0.0417 \\ 0.6891 & 0.1356 & 0.0551 & 0.0433 & 0.0769 \\ 0.0330 & 0.1819 & 0.0764 & 0.0387 & 0.6700 \\ 0.0474 & 0.0559 & 0.0676 & 0.7176 & 0.1115 \end{bmatrix}$$

$$\pi^{(2)}_{ij|1} = \begin{bmatrix} 0.0207 & 0.0231 & 0.0248 & 0.1097 & 0.8217 \\ 0.1596 & 0.3102 & 0.1873 & 0.0354 & 0.3076 \\ 0.3395 & 0.1357 & 0.1502 & 0.3434 & 0.0312 \\ 0.0476 & 0.7906 & 0.0511 & 0.0754 & 0.0353 \\ 0.7336 & 0.1364 & 0.0592 & 0.0250 & 0.0458 \end{bmatrix}$$

$$\pi^{(2)}_{ij|2} = \begin{bmatrix} 0.5675 & 0.1255 & 0.0879 & 0.0695 & 0.1497 \\ 0.6323 & 0.0172 & 0.0413 & 0.2666 & 0.0425 \\ 0.4100 & 0.4899 & 0.0042 & 0.0545 & 0.0414 \\ 0.2143 & 0.2458 & 0.1916 & 0.0003 & 0.3481 \\ 0.0283 & 0.0763 & 0.0169 & 0.0306 & 0.8479 \end{bmatrix}$$

$$\pi^{(3)}_{ij|1} = \begin{bmatrix} 0.4275 & 0.0147 & 0.2877 & 0.2059 & 0.0641 \\ 0.0698 & 0.0456 & 0.8262 & 0.0036 & 0.0549 \\ 0.1302 & 0.4898 & 0.2003 & 0.0524 & 0.1274 \\ 0.8679 & 0.0810 & 0.0095 & 0.0113 & 0.0303 \\ 0.1211 & 0.3233 & 0.5139 & 0.0277 & 0.0140 \end{bmatrix}$$

$$\pi^{(3)}_{ij|2} = \begin{bmatrix} 0.9948 & 0.0036 & 0.0009 & 0.0002 & 0.0005 \\ 0.1862 & 0.0161 & 0.2739 & 0.4994 & 0.0245 \\ 0.2133 & 0.2234 & 0.1280 & 0.1845 & 0.2508 \\ 0.1256 & 0.0539 & 0.2035 & 0.2515 & 0.3655 \\ 0.0001 & 0.0000 & 0.0005 & 0.9994 & 0.0000 \end{bmatrix}$$

$$\pi^{(4)}_{ij|1} = \begin{bmatrix} 0.9988 & 0.0005 & 0.0003 & 0.0003 & 0.0001 \\ 0.2055 & 0.2464 & 0.0253 & 0.3412 & 0.1816 \\ 0.5338 & 0.0025 & 0.3090 & 0.1541 & 0.0005 \\ 0.0392 & 0.0072 & 0.0236 & 0.0287 & 0.9015 \\ 0.0110 & 0.0387 & 0.0028 & 0.0241 & 0.9235 \end{bmatrix}$$

$$\pi^{(4)}_{ij|2} = \begin{bmatrix} 0.4398 & 0.3101 & 0.0833 & 0.0901 & 0.0767 \\ 0.3459 & 0.1314 & 0.3912 & 0.0804 & 0.0511 \\ 0.0009 & 0.0004 & 0.0004 & 0.0069 & 0.9915 \\ 0.1971 & 0.2796 & 0.2147 & 0.2461 & 0.0625 \\ 0.1667 & 0.0280 & 0.0771 & 0.2424 & 0.4858 \end{bmatrix}$$

and the individual utility matrices be defined by

$$J^{(1)}_{ij|1} = \begin{bmatrix} 86 & 8 & 53 & 58 & 9 \\ 57 & 9 & 48 & 5 & 63 \\ 3 & 1 & 81 & 4 & 67 \\ 0 & 87 & 23 & 59 & 73 \\ 59 & 62 & 50 & 25 & 90 \end{bmatrix}$$

$$J^{(1)}_{ij|2} = \begin{bmatrix} 756 & 252 & 156 & 180 & 6360 \\ 108 & 360 & 312 & 768 & 624 \\ 2040 & 396 & 48 & 576 & 480 \\ 456 & 648 & 444 & 396 & 156 \\ 6000 & 696 & 2880 & 768 & 504 \end{bmatrix}$$

$$J^{(2)}_{ij|1} = \begin{bmatrix} 12 & 94 & 100 & 14 & 48 \\ 16 & 74 & 8 & 4 & 6 \\ 18 & 74 & 80 & 94 & 4 \\ 64 & 8 & 52 & 32 & 86 \\ 58 & 88 & 18 & 30 & 56 \end{bmatrix}$$

$$J^{(2)}_{ij|2} = \begin{bmatrix} 24 & 31 & 15 & 43 & 8 \\ 58 & 41 & 27 & 12 & 40 \\ 87 & 84 & 9 & 5 & 1 \\ 41 & 41 & 43 & 71 & 23 \\ 12 & 40 & 26 & 25 & 1 \end{bmatrix}$$

$$J^{(3)}_{ij|1} = \begin{bmatrix} 1 & 1 & 0 & 3 & 11 \\ 1 & 4 & 2 & 24 & 9 \\ 2 & 3 & 16 & 28 & 23 \\ 6 & 1 & 23 & 17 & 1 \\ 3 & 9 & 19 & 4 & 2 \end{bmatrix}$$

$$J^{(3)}_{ij|2} = \begin{bmatrix} 8 & 2 & 6 & 3 & 5 \\ 3 & 0 & 2 & 8 & 6 \\ 16 & 5 & 3 & 28 & 14 \\ 3 & 6 & 3 & 6 & 7 \\ 2 & 22 & 3 & 9 & 0 \end{bmatrix}$$

$$J^{(4)}_{ij|1} = \begin{bmatrix} 5 & 11 & 2 & 5 & 6 \\ 9 & 5 & 3 & 1 & 4 \\ 5 & 3 & 2 & 7 & 1 \\ 0 & 1 & 8 & 5 & 3 \\ 0 & 2 & 0 & 1 & 4 \end{bmatrix}$$

$$J^{(4)}_{ij|2} = \begin{bmatrix} 0 & 11 & 3 & 4 & 5 \\ 1 & 2 & 13 & 1 & 6 \\ 2 & 7 & 1 & 7 & 6 \\ 0 & 1 & 8 & 7 & 6 \\ 1 & 9 & 6 & 9 & 4 \end{bmatrix}$$

Remark 12. The Stackelberg is an asynchronous game. We first compute the two leaders given the followers. Then, we compute the followers given the leaders. Because, the practical application of the problem involves the computation of two leaders and the two followers we can makes use of quadratic programming. If the problem is extended to more players it is evidently can not be interpreted as a QP problem: in this case must be used a nonlinear programming method design for poly linear cost functions and constraints [38].

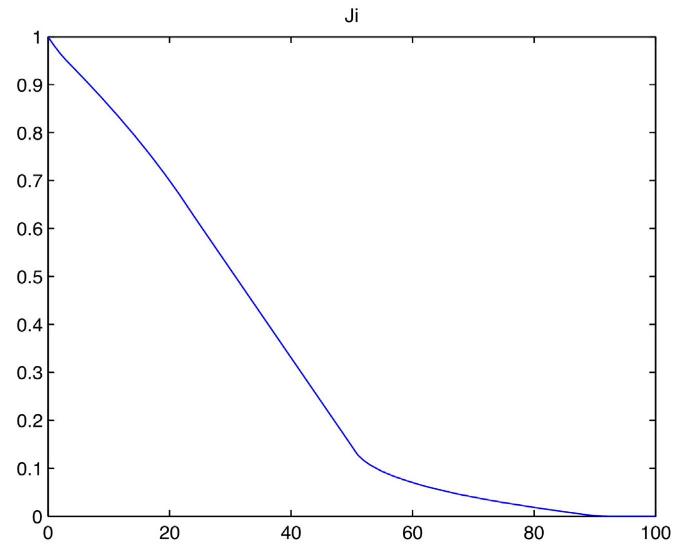

Fig. 1. Convergence of the Ji parameter.

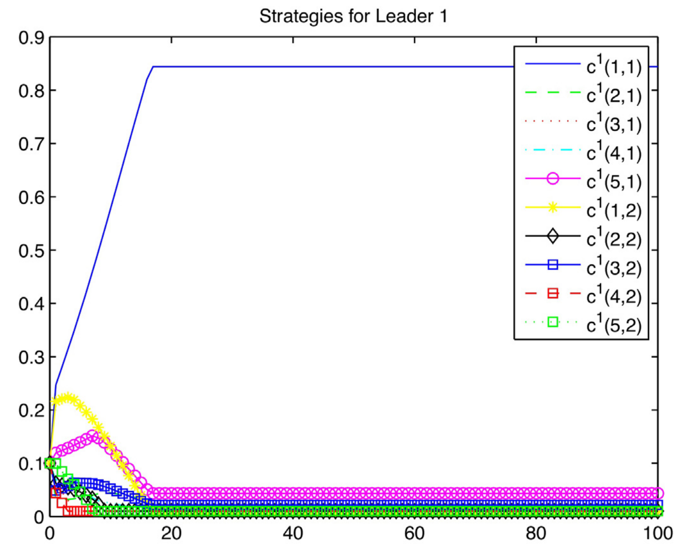

Fig. 3. Convergence of the strategies of the defender 1.

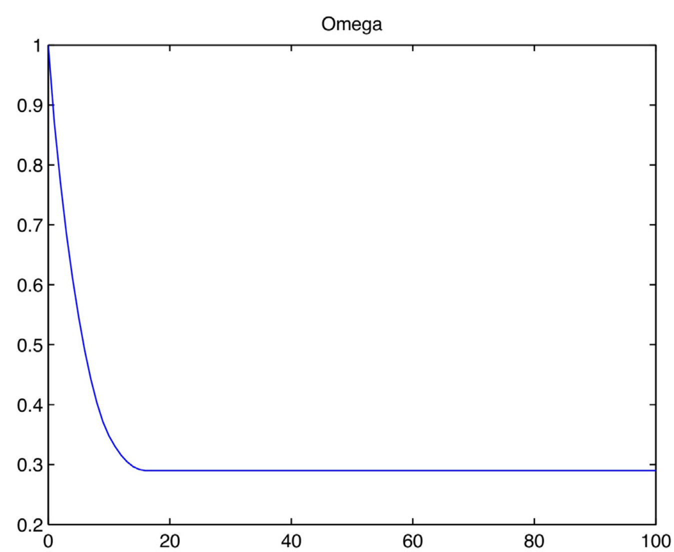

Fig. 2. Convergence of the Omega parameter.

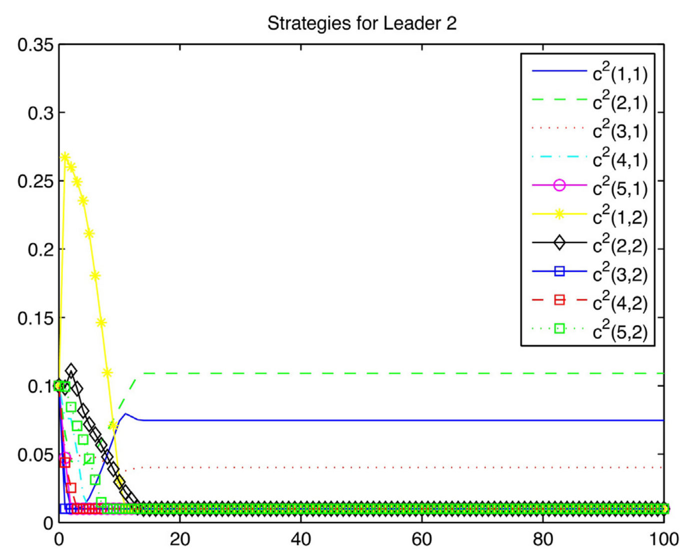

Fig. 4. Convergence of the strategies of the defender 2.

Figs. 1 and 2 show the Ji and Omega parameters. Figs. 3–6 show the convergence of the strategies in terms of the variable c. With final $\lambda^{(1)^*}=0.2510$ and $\lambda^{(1)^*}=0.7490$ for the defenders (see Fig. 7), the mixed strategies obtained for determining the strong Stackelberg/Nash equilibrium for all the players applying (8) are as follows

$$d^{(1)*}=\begin{bmatrix}0.9883 & 0.0117\\ 0.7488 & 0.2512\\ 0.3145 & 0.6855\\ 0.5208 & 0.4792\\ 0.8122 & 0.1878\end{bmatrix}\quad d^{(2)*}=\begin{bmatrix}0.8821 & 0.1179\\ 0.9160 & 0.0840\\ 0.8013 & 0.1987\\ 0.1931 & 0.8069\\ 0.0144 & 0.9856\end{bmatrix}\quad d^{(3)*}$$

$$=\begin{bmatrix}0.9755 & 0.0245\\ 0.8736 & 0.1264\\ 0.0474 & 0.9526\\ 0.9526 & 0.0474\\ 0.1097 & 0.8903\end{bmatrix}\quad d^{(4)*}=\begin{bmatrix}0.9492 & 0.0508\\ 0.6328 & 0.3672\\ 0.5000 & 0.5000\\ 0.3274 & 0.6726\\ 0.1623 & 0.8377\end{bmatrix}$$

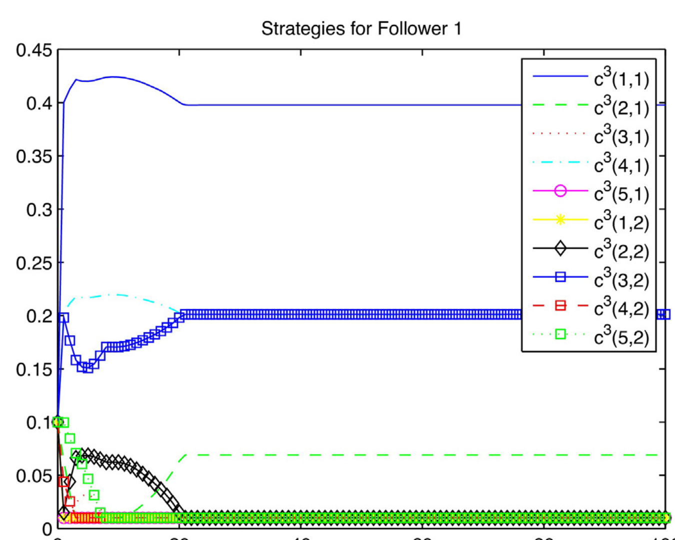

Fig. 5. Convergence of the strategies of the attacker 1.

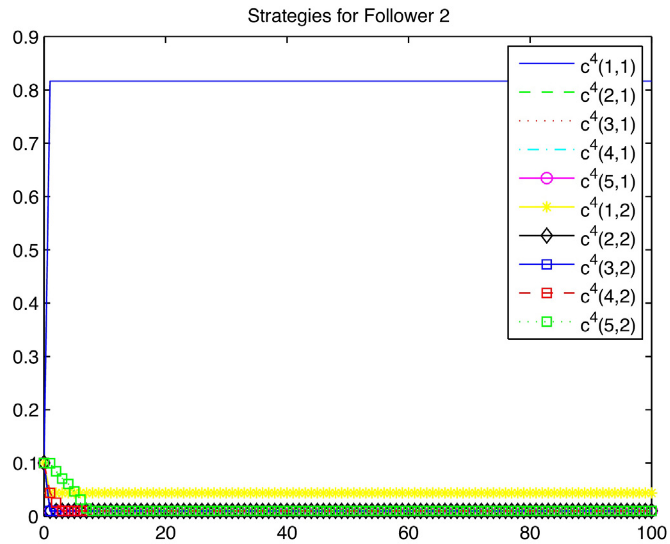

Fig. 6. Convergence of the strategies of the attacker 2.

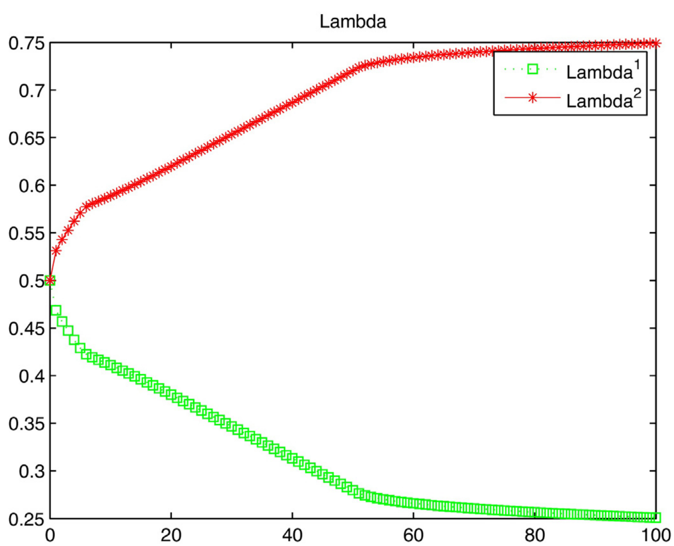

Fig. 7. Convergence of lambda.

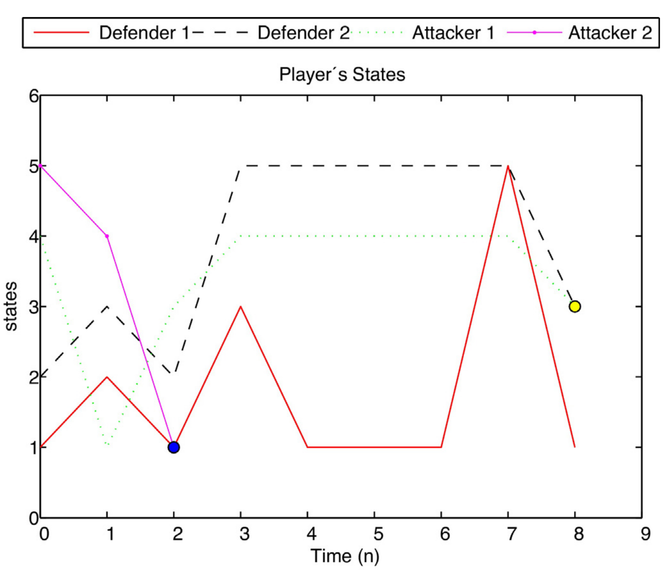

Fig. 8. Patrol schedule – scenario 1.

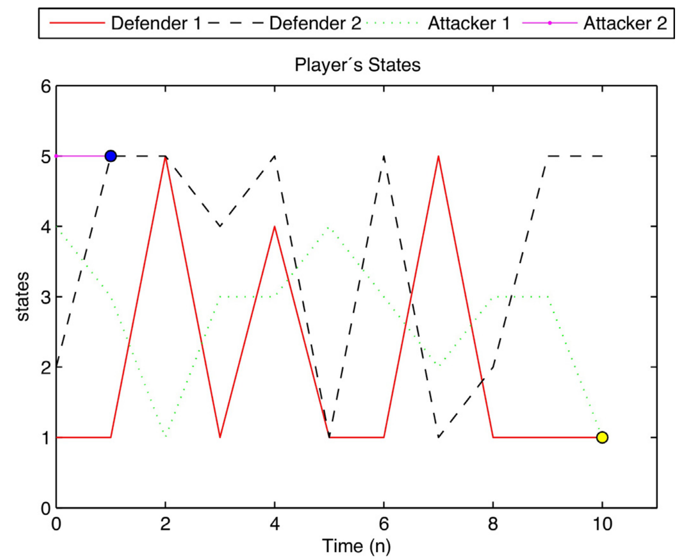

Fig. 9. Patrol schedule – scenario 2.

Strategies $d^{(1)^*}$ and $d^{(2)^*}$ correspond to the defender 1 and defender 2, respectively. As well as, strategies $d^{(3)^*}$ and $d^{(4)^*}$ correspond to the attacker 1 and attacker 2, respectively.

For the Patrol Schedule model of the security game we define the initial state i of each player as: follows $i^{(1)}(0)=1$, $i^{(2)}(0)=2$, $i^{(3)}(0)=4$, $i^{(4)}(0)=5$. The numerical example proves that the sequences generated by the two random walks of both, defenders and attackers, converge to the product of the individual probabilities. As a result, for a first scenario (see Fig. 8) we obtain that the attacker 2 is caught at state 1 after 2 steps by the defender 1 and attacker 1 is caught at state 3 after 8 steps by the defender 2, so the game is over. In a second scenario (see Fig. 9), the attacker 2 is caught at state 5 after 1 step by the defender 2 and attacker 1 is caught at state 1 after 10 steps also by the defender 1, so the game is over.

7. Conclusions

The reliable patrolling is a complex problem, requiring solutions that integrate efficiency and unpredictability. To address this shortcoming, this paper presented four main contributions.

- First, we presented a model for conforming the coalition of the defenders in case of a metric state space computing the Strong L_p-Nash equilibrium.
- Second, we solved the problem of computing Stackelberg/Nash equilibrium for this game using the extraproximal method.
- Third, we developed the method for the Stackelberg game in terms of nonlinear programming problems implementing the Lagrange principle and employed Tikhonov's regularization method to ensure the convergence of the cost-functions to a unique Stackelberg/Nash equilibrium.
- Fourth, we developed a random walk where actions are selected using the max operator. We proved that the synchronization of the random walk of defenders and attackers converge in probability to the product of the individual probabilities.

Finally, we applied our approach in a numerical example showing the efficiency of the solution.

References

[1] M. Tambe, Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned, Cambridge University Press, 2011.

[2] J. Pita, M. Jain, F. Ordoñez, C. Portwa, M. Tambe, C. Western, Using game theory for Los Angeles airport security, AI Mag. 30 (1) (2009) 43–57.
[3] M. Jain, E. Kardes, C. Kiekintveld, F. Ordo nez, M. Tambe, Security games with arbitrary schedules: a branch and price approach, in: Proceedings of the National Conference on Artificial Intelligence (AAAI), Atlanta, GA, USA, 2010.
[4] J. Tsai, S. Rathi, C. Kiekintveld, F. Ordo nez, M. Tambe, IRIS – a tool for strategic security allocation in transportation networks, in: Eighth International Conference on Autonomous Agents and Multiagent Systems – Industry Track, 2009, pp. 37–44.
[5] M. Jain, C. Kiekintveld, M. Tambe, Quality-bounded solutions for finite Bayesian Stackelberg games: scaling up, in: Proceedings of the Tenth International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS), Taipei, Taiwan, 2011, pp. 997–1004.
[6] B. An, J. Pita, E. Shieh, M. Tambe, C. Kiekintveld, J. Marecki, Guards and protect: next generation applications of security games, SIGECOM 10 (2011) 31–34.
[7] J. Pita, M. Tambe, C. Kiekintveld, S. Cullen, E. Steigerwald, Guards – game theoretic security allocation on a national scale, in: Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 2011.
[8] E. Shieh, B. An, R. Yang, M. Tambe, C. Baldwin, J. DiRenzo, B. Maule, G. Meyer, Protect: a deployed game theoretic system to protect the ports of the United states, in: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems, 2012.
[9] F. Fang, A. Jiang, M. Tambe, Protecting moving targets with multiple mobile resources, J. Artif. Intell. Res. 48 (2013) 583–634.
[10] P. Varakantham, H.C. Lau, Z. Yuan, Scalable randomized patrolling for securing rapid transit networks, in: Proceedings of the Conference for Innovative Applications for Artificial Intelligence (IAAI), Bellevue, Washington, USA, 2013, pp. 1563–1568.
[11] J.B. Clempner, A.S. Poznyak, Stackelberg security games: computing the shortest-path equilibrium, Expert Syst. Appl. 42 (8) (2015) 3967–3979.
[12] K.K. Trejo, J.B. Clempner, A.S. Poznyak, A Stackelberg security game with random strategies based on the extraproximal theoretic approach, Eng. Appl. Artif. Intell. 37 (2015) 145–153.
[13] R. Yang, C. Kiekintveld, F. Ordonez, M. Tambe, R. John, Improving resource allocation strategy against human adversaries in security games, in: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), Barcelona, Catalonia, Spain, 2011, pp. 458–464.
[14] B. An, M. Tambe, F. Ordonez, E. Shieh, C. Kiekintveld, Refinement of strong Stackelberg equilibria in security games, in: Proceedings of the Twenty-fifth Conference for the Advancement of Artificial Intelligence (AAAI), San Francisco, CA, USA, 2011, pp. 587–593.
[15] B. An, D. Kempe, C. Kiekintveld, E. Shieh, S. Singh, M. Tambe, Security games with limited surveillance, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), Toronto, Ontario, Canada, 2012, pp. 1241–1247.
[16] A. Blum, N.a. Haghtalab, A.D. Procaccia, Lazy defenders are almost optimal against diligent attackers, in: Proceedings of the 28th AAAI Conference on Artificial Intelligence, Québec, Canada, 2014, pp. 573–579.
[17] J. Letchford, Y. Vorobeychik, Optimal interdiction of attack plans, in: Proceedings of the Twelfth International Conference of Autonomous Agents and Multi-agent Systems (AAMAS), Saint Paul, MN, USA, 2013, pp. 199–206.
[18] Z. Yin, M. Tambe, A unified method for handling discrete and continuous uncertainty in Bayesian Stackelberg games, in: Proceedings of the Eleventh International Conference on Autonomous Agents and Multiagent Systems (AAMAS), Valencia, Spain, 2012, pp. 234–242.
[19] A. Jiang, Z. Yin, C. Zhang, M. Tambe, S. Kraus, Game-theoretic randomization for security patrolling with dynamic execution uncertainty, in: Proceedings of the International Conference on Autonomous Agents and Multi-agent Systems (AMAAS), Saint Paul, MN, USA, 2013, pp. 207–214.
[20] C. Kiekintveld, T. Islam, V. Kreinovich, Security games with interval uncertainty, in: Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Saint Paul, MN, USA, 2013, pp. 231–238.
[21] D. Korzhyk, Z. Yin, C. Kiekintveld, V. Conitzer, M. Tambe, Stackelberg vs. Nash in security games: an extended investigation of interchangeability, equivalence, and uniqueness, J. Artif. Intell. Res. 41 (2) (2011) 297–327.
[22] N. Basilico, N. Gatti, F. Villa, Asynchronous multi-robot patrolling against intrusions in arbitrary topologies, in: Proceeding of the Conference for the Advancement of Artificial Intelligence (AAAI), Atlanta, GA, USA, 2010, pp. 345–350.
[23] N. Agmon, G.A. Kaminka, S. Kraus, Multi-robot adversarial patrolling: facing a full-knowledge opponent, J. Artif. Intell. Res. 42 (1) (2011) 887–916.
[24] E. Hernández, A. Barrientos, J. del Cerro, Selective smooth fictitious play: an approach based on game theory for patrolling infrastructures with a multi-robot system, Expert Syst. Appl. 41 (6) (2014) 2897–2913.
[25] S. Shen, Han, L. Guo, W. Li, Q. Cao, Survivability evaluation towards attacked WSNs based on stochastic game and continuous-time Markov chain, Appl. Soft Comput. (2012) 1467–1476.
[26] A.N. Tikhonov, V.Y. Arsenin, Solution of Ill-posed Problems, Winston & Sons, Washington, 1977.
[27] A. Tikhonov, A. Goncharsky, V. Stepanov, A.G. Y, Numerical Methods for the Solution of Ill-Posed Problems, Kluwer Academic Publishers, 1995.
[28] A.S. Antipin, An extraproximal method for solving equilibrium programming problems and games, Comput. Math. Math. Phys. 45 (11) (2005) 1893–1914.
[29] K.K. Trejo, J.B. Clempner, A.S. Poznyak, Computing the Stackelberg/Nash equilibria using the extraproximal method: convergence analysis and implementation details for Markov chains games, Int. J. Appl. Math. Comput. Sci. 25 (2) (2015) 337–351.
[30] J.B. Clempner, A.S. Poznyak, Simple computing of the customer lifetime value: a fixed local–optimal policy approach, J. Syst. Sci. Syst. Eng. 23 (4) (2014) 439–459.
[31] A.S. Poznyak, K. Najim, E. Gomez-Ramirez, Self-learning Control of Finite Markov Chains, Marcel Dekker, New York, 2000.
[32] G. Eichfelder, Adaptive Scalarization Methods in Multiobjective Optimization, Springer Science & Business Media, 2008.
[33] J.B. Clempner, A.S. Poznyak, Solving the Pareto front for multiobjective Markov chains using the minimum Euclidean distance gradient-based optimization method, Math. Comput. Simul. 119 (2015) 142–160.
[34] Y. Germeyer, Introduction to the Theory of Operations Research, Nauka, Moscow, 1971.
[35] Y. Germeyer, Games with Nonantagonistic Interests, Nauka, Moscow, 1976.
[36] K.K. Trejo, J.B. Clempner, A.S. Poznyak, An optimal strong equilibrium solution for cooperative multi-leader-follower Stackelberg Markov chains games, Kibernetika 52 (2) (2016) 258–279.
[37] A.S. Poznyak, Advance Mathematical Tools for Automatic Control Engineers Deterministic Techniques, vol. 2, Elsevier, Amsterdam, 2009.
[38] S. Cafieri, P. Hansen, L. Létocart, L. Liberti, F. Messine, Compact Relaxations for Polynomial Programming Problems Experimental Algorithms, Lecture Notes in Computer Science, vol. 7276, Springer, Berlin, Heidelberg, 2012, pp. 75–86.