MODELING SHORTEST PATH GAMES WITH PETRI NETS: A LYAPUNOV BASED THEORY

JULIO CLEMPNER

Center for Computing Research
National Polytechnic Institute
Av. Juan de Dios Batiz s/n, Edificio CIC
Col. Nueva Industrial Vallejo, 07738
Mexico City, Mexico
e-mail: julio@k-itech.com

In this paper we introduce a new modeling paradigm for shortest path games representation with Petri nets. Whereas previous works have restricted attention to tracking the net using Bellman's equation as a utility function, this work uses a Lyapunov-like function. In this sense, we change the traditional cost function by a trajectory-tracking function which is also an optimal cost-to-target function. This makes a significant difference in the conceptualization of the problem domain, allowing the replacement of the Nash equilibrium point by the Lyapunov equilibrium point in game theory. We show that the Lyapunov equilibrium point coincides with the Nash equilibrium point. As a consequence, all properties of equilibrium and stability are preserved in game theory. This is the most important contribution of this work. The potential of this approach remains in its formal proof simplicity for the existence of an equilibrium point.

Keywords: shortest path game, game theory, Nash equilibrium point, Lyapunov equilibrium point, Bellman's equation, Lyapunov-like fuction, stability

1. Introduction

Decision making (Bellman, 1957; Howard, 1960; Puterman, 1994) is the study of identifying and choosing strategies which guide the selection of actions that lead a decision maker to the final decision state. Decisions involve a choice based on a set of predefined criteria and the identification of alternatives. The available alternatives influence the criteria we apply to them, and, similarly, the criteria we establish influence the alternatives we will consider.

The optimal control of such systems needs to take into account different circumstances and preferences and, in addition, the treatment of the uncertainty of future events (Hernández-Lerma, 1996; Hernández-Lerma and Lasserre, 1999). In most real applications, the optimum performance is an impossible goal. On the contrary, certain problems present a particular structure that allows us to avoid complexity and to achieve optimality efficiently. This creates the need for extending the decision-making framework to make good decisions efficiently finding the shortest-path and the minimum cost-to-target.

Markov decision processes are commonly used to analyze shortest-path and minimum cost-to-target problems in which a natural form of termination guarantees that the expected future costs are bounded, at least under some policies. The stochastic shortest path problem (Bertsekas and Shreve, 1978; Bertsekas, 1987; Blackwell, 1967; Derman, 1970; Kushner, 1971; Strauch, 1966; Whittle, 1983) is a generalization through which each node has a probability distribution over all possible successor nodes. Given a starting node and a selection of distributions, we wish that the path led to a final point with probability one having the minimum expected length. Note that if it assigned a probability one to every probability distribution of each single successor node, then a deterministic shortest path problem is obtained.

In this sense, finite action-state and action-transient Markov decision processes with positive cost functions were first formulated and studied by Eaton and Zadeh (1962). They called it a *problem of pursuit* that consists of intercepting in a minimum expected time a target that moves randomly among a finite number of states. In the study, they established the idea of a proper policy and supposed that at each state, except the final state, the set of controls is finite. Pallu de la Barriere (1967) supported and improved these results. Derman (1970) also extended these results under the title of *first passage problems*, observing that the finite-state Markovian decision problem is a particular case. Veinott (1969) obtained similar results to that of Eaton and Zadeh (1962) proving that dynamic programming mapping is a contraction under the assumption that all stationary policies are proper (transient). Kushner (1971) enhanced the results of Eaton and Zadeh (1962) letting the set of controls to be infinite at each state, and restricting the state space with a compactness assumption. Whittle (1983), under the name *transient programming*, supported the results obtained by Veinott (1969). He extended the problem presented by Veinott to an infinite state and control spaces under uniform boundedness conditions on the expected termination time.

Bertsekas and Tsistsiklis (Bertsekas and Shreve, 1978; Bertsekas, 1987; Bertsekas and Tsitsiklis, 1989), denoting the problem as the *stochastic shortest path problem*, improved the results of Eaton and Zadeh (1962); Veinott (1969) and Whittle (1983), by weakening the condition that all policies be transient. They established that every stationary deterministic policy can have a value function associated, which is unbounded from above. Bertsekas and Tsistsiklis (1991), in a subsequent work, strengthened their previous result by relaxing the condition that the set of actions available in each state be finite. They assumed that the set of actions available in each state is compact, the transition kernel is continuous over the set of actions available in each state, and the cost function is semi-continuous (over the set of actions available in each state) and bounded. Hinderer and Waldmann (2003; 2005) improved the result presented by Mandl (1969), Veinott (1969) and Rieder (1975) for finite Markovian decision processes with an absorbing set. They were interested in the critical discount factor, defined as the smallest number such that for all discount factors β smaller than this value, the limit v of the n-stage β-discounted optimal value function exists and is finite for each choice of the one-stage reward function. Pliska (1978) assumed that the cost function was bounded and that all policies were transient, additionally to the well-known assumptions of a compact action space, a continuous transition kernel, and a lower semi-continuous cost function. It is important to note that this work was the first one to extend the problem to Borel states and action spaces. Hernandez-Lerma *et al.* (1999) expanded the results of Pliska by weakening the condition that the cost function was bounded and supposing that it was dominated by a given function.

The optimal stopping problem is directly related to the stochastic shortest-path problem and was investigated by Dynkin (1963), and Grigelionis and Shiryaev (1966), and considered extensively in the literature by others (Derman, 1970; Kushner, 1971; Shiryaev, 1978; Whittle, 1983). It is a special type of the transient Markov decision process where a state-dependent cost is incurred only when invoking a stopping action which leads the system to the destination (finish); all costs are zero before stopping. For optimal stopping problems, the associated value function of the policy (under which the stopping action is never taken) is equal to zero at all states: however, it is not transient.

Shortest path games are usually conceptualized as two-player, zero-sum games. On the one hand, the "minimizing" player seeks to drive a finite-state dynamic system to reach a terminal state along the least expected cost path. On the other hand, the "maximizer" player seeks to maximize the expected total cost interfering with the minimizer's progress. In playing the game, the players implement actions simultaneously at each state, with a full knowledge of the state of the system but without any knowledge of each other's current decision.

Shapley (1953) provided the first work on shortest path games. In his paper, two players are successively faced with matrix-games of mixed strategies where both the immediate cost and transition probabilities to new matrix-games are influenced by the decisions of the players. In this conceptualization, the state of the system is the matrix-game currently being played. Kushner and Chamberlain (1969) took into account undiscounted, pursuit, stochastic games. They assumed that the state space is finite with a final state corresponding to the evader being trapped, and they considered pure strategies over compact action spaces. Under these considerations, they proved that there exists an equilibrium cost vector for the game which can be found through value iteration. Van der Wal (1981) explored a particular case of the research by Kushner and Chamberlain (1969), producing error bounds for updates in the value iteration, considering restrictive assumptions about the capability of the pursuer to capture the evader. Kumar and Shiau (1981), for the case of a non-negative additive cost, proved the existence of an extended real equilibrium cost vector in non-Markov randomized policies. They showed that the minimizing player can achieve the equilibrium using a stationary Markov randomized policy. In addition, for the case where the state space is finite, the maximizing player can play ε-optimally using stationary randomized policies.

Patek and Bertsekas (Patek, 1997; Patek and Bertsekas, 1999) analyzed the case of two players, where one player seeks to drive the system to termination along a least cost path and the other seeks to prevent the termination altogether. They did not assume the non-negativity of the costs, and the analysis, being much more complicated than the corresponding analysis of Kushner and Chamberlain (1969), generalized (to the case of two players) those for stochastic shortest path problems (Bertsekas and Tsitsiklis, 1991). Patek and Bertsekas proposed alternative assumptions which guarantee that, at least under optimal policies, the terminal state is reached with probability one. They considered undiscounted additive cost games without averaging, admitting that there are policies for the minimizer which allow the maximizer to prolong the game indefinitely at an infinite cost to the minimizer. Under assumptions which generalize deterministic shortest path problems, they established (i) the existence of a real-valued equilibrium cost vector achievable with stationary policies for the opposing players and (ii) the convergence of the value iteration and the policy iteration to the unique solution of Bellman's equation (Bellman, 1957). The results of Patek and Bertsekas did imply the results of Shapley (1953), as well as those of Kushner and Chamberlain (1969). Because of their assumptions relating to termination, they were able to derive conclusions stronger than those made by Kumar and Shiau (1981) for the case of a finite state space. In a subsequent work, Patek (2001) reexamined the stochastic shortest path formulation in the context of Markov decision processes with an exponential utility function.

Whereas previous works have restricted attention to track the net using Bellman's equation as a utility function (Bellman, 1957), this paper introduces a modeling paradigm for developing a decision process representation called the Game Petri Net (GPN). The idea is to use a *trajectory function* that is non-negative and converges in an optimal way to the equilibrium point. In this equilibrium, each player chooses a strategy with a trajectory value equal to the value that this strategy is a best reply to a strategy profile chosen by the opponents. The advantage of this approach is that fixed-point conditions for the game are given by the definition of the Lyapunov-like function: however, formally it is not necessary for a fixed-point theorem to satisfy Nash equilibrium conditions as usual. In addition, new properties of equilibrium and stability (Clempner, 2005) are consequently introduced for finite *n*-player games.

The game Petri net extends the place-transitions Petri net theoretic approach including Markov decision processes, using a trajectory-tracking function as a tool for path planning. Although both perspectives are integrated in a GPN, they work at different execution levels. That is, the operation of the place-transition Petri net is not modified and the trajectory function is used exclusively for establishing a trajectory tracking in the place-transition Petri net.

In the GPN we introduce the well-known Nash equilibrium point concept. We also introduce an alternative definition to the Nash equilibrium point that we call the steady-state equilibrium point in the sense of Lyapunov. The steady-state equilibrium point is represented in the GPN by the optimum point. We show that the optimum point (the steady-state equilibrium point) and the Nash equilibrium point coincide. It is interesting to note that the steady-state equilibrium point lends necessary and sufficient conditions of stability to the game (Clempner, 2005).

The paper is structured in the following manner: The next section presents the necessary mathematical background and terminology needed to understand the rest of the paper. Section 3 describes the GPN, and all structural assumptions are introduced. Section 4 discusses the main results of the paper, giving a detailed analysis of equilibrium conditions for the GPN. Finally, in Section 4 some concluding remarks are provided.

2. Preliminaries

In this section, we present some well-established definitions and properties which will be used later.

Notation 1. $\mathbb{N} = \{0, 1, 2, \dots\}$, $\mathbb{R}_+ = [0, \infty)$, $\mathbb{N}_{n_0+} = \{n_0, n_0+1, \dots, n_0+k, \dots\}$, $n_0 \geq 0$. We represent by $\bar{\mathbf{0}}$ the vector $(0, \dots, 0) \in \mathbb{R}^d$ and by $\bar{\mathbf{C}}$ the vector of constants $(C, \dots, C) \in \mathbb{R}^d$. Given $x, y \in \mathbb{R}^d$, we usually denote the relation "$\leq$" to mean componentwise inequalities with the same relation, i.e., $x \leq y$ is equivalent to $x_i \leq y_i, \forall i$. A function $f(n, x)$, $f : \mathbb{R}^d \to \mathbb{R}^d$ is called nondecreasing in x if, given $x, y \in \mathbb{R}^d$ such that $x \geq y$ and $n \in \mathbb{N}_{n_0+}$, we have $f(n, x) \geq f(n, y)$.

2.1. Petri Nets.

Petri nets are a tool for the study of systems. Petri net theory allows a system to be modeled by a Petri net, a mathematical representation of the system. The analysis of the Petri net then can, hopefully, reveal important information about the structure and dynamic behavior of the modeled system. This information can then be used to evaluate the modeled system and to suggest improvements or changes.

A Petri net is the quintuple $PN = \{P, Q, F, W, M_0\}$, where $P = \{p_1, p_2, \dots, p_m\}$ is a finite set of places, $Q = \{q_1, q_2, \dots, q_n\}$ is a finite set of transitions, $F \subseteq (P \times Q) \cup (Q \times P)$ is a set of arcs, $W : F \to \mathbb{N}_1^+$ is a weight function, M_0: $P \to \mathbb{N}$ is the initial marking, $P \cap Q = \varnothing$ and $P \cup Q \neq \varnothing$.

A Petri net structure without any specific initial marking is denoted by N. A Petri net with the given initial marking is denoted by (N, M_0). Notice that if $W(p, q) = \alpha$ (or $W(q, p) = \beta$), then this is often represented graphically by α, (β) arcs from p to q (q to p) each with no numeric label.

Let $M_k(p_i)$ denote the marking (i.e., the number of tokens) at the place $p_i \in P$ at the time k, and let $M_k = [M_k(p_1), \dots, M_k(p_m)]^T$ denote the marking (state) of PN at the time k. A transition $q_j \in Q$ is said to be enabled at the time k if $M_k(p_i) \geq W(p_i, q_j)$ for all $p_i \in P$ such that $(p_i, q_j) \in F$. It is assumed that at each time k, there exists at least one transition to fire, i.e., it is not possible to block the net. If a transition is enabled, then it can fire. If an enabled transition $q_j \in Q$ fires at the time k, then the next marking for $p_i \in P$ is given by

$$M_{k+1}(p_i) = M_k(p_i) + W(q_j, p_i) - W(p_i, q_j).$$

Let $A = [a_{ij}]$ denote an $n \times m$ matrix of integers (the incidence matrix), where $a_{ij} = a_{ij}^+ - a_{ij}^-$ with $a_{ij}^+ =$

$W(q_i, p_j)$ and $a^-_{ij} = W(p_j, q_i)$. Let $u_k \in \{0,1\}^n$ denote a firing vector, where if $q_j \in Q$ is fired, then its corresponding firing vector is $u_k = [0,\dots,0,1,0,\dots,0]^T$ with "1" in the j-th position in the vector and zeros everywhere else. The matrix equation (non-linear difference equation) describing the dynamical behavior represented by a Petri net is

$$M_{k+1} = M_k + A^T u_k, \qquad (1)$$

where if at the step k, $a^-_{ij} < M_k(p_j)$ for all $p_j \in P$, then $q_i \in Q$ is enabled, and if this $q_i \in Q$ fires, then its corresponding firing vector u_k is utilized in the difference equation (1) to generate the next step. Notice that if M' can be reached from some other marking M, and if we fire some sequence of d transitions with corresponding firing vectors $u_0, u_1, \dots, u_{d-1}$, we obtain

$$M' = M + A^T u, \quad u = \sum_{k=0}^{d-1} u_k. \qquad (2)$$

Definition 1. The set of all markings (states) reachable from some starting marking M is called the *reachability set*, and is denoted by $R(M)$.

Let $(\mathbb{N}_{n_0+}, d)$ be a metric space where $d : \mathbb{N}_{n_0+} \times \mathbb{N}_{n_0+} \to \mathbb{R}_+$ is defined by

$$d(M_1, M_2) = \sum_{i=1}^{m} \zeta_i \mid M_1(p_i) - M_2(p_i) \mid, \qquad \zeta_i > 0, \quad i = 1,\dots,m.$$

2.2. Bellman's Equation. We assume that every discrete event system with a finite set of states P to be controlled can be described as a fully-observable, discrete-state Markov decision process (Bellman, 1957; Howard, 1960; Puterman, 1994). To control the Markov chain, there must exist a possibility of changing the probability of transitions through an external interference. We suppose that there exists a possibility to carry out the Markov process by N different methods. In this sense, we suppose that the controlling of the discrete event system has a finite set of actions Q available which cause stochastic state transitions. We denote by $p_q(s,t)$ the probability that action q generates a transition from the state s to the state t where $s, t \in P$.

A stationary policy $\pi : P \to Q$ denotes a particular strategy or a course of action to be adopted by a discrete event system, with $\pi(s,q)$ being the action to be executed whenever the discrete event system is in the state $s \in P$. We refer to Bellman (1957), Howard (1960) and Puterman (1994) for a description of policy construction techniques.

Hereafter, we will consider having the possibility to estimate every step of the process through a utility function that represents the utility generated by the transition from the state s to the state t in the case of using an action q. We assume an infinite time horizon, and that the discrete event system accumulates the utility associated with the states it enters.

Let us define $V_\pi(s)$ as the maximum utility starting at the state s that guarantees choosing the optimal course of action $\pi(s,q)$. Let us suppose that at the state s we have an accumulated utility $R(s)$ and the previous transitions have been executed in an optimal form. In addition, let us assume that the transition of going from the state s to the state t has a probability of $p_{\pi(s,q)}(s,t)$. Because the transition from the state s to the state t is stochastic, it is necessary to take into account the possibility of going through all the possible states from s to t. Then the utility of going from the state s to state t is represented by Bellman's equation (Bellman, 1957):

$$V_\pi(s) = R(s) + \beta \sum_{t \in P} p_{\pi(s,q)}(s,t) V_\pi(t), \qquad (3)$$

where $\beta \in [0,1)$ is the discount rate (Howard, 1960).

The value of π at any initial state s can be computed by solving this system of linear equations. A policy π is optimal if $V_\pi(t) \geq V_{\pi'}(t)$ for all $t \in P$ and policies π'. The function V establishes a preference relation.

3. Game Petri Nets

The aim of this section is to associate to any shortest path game a game Petri net (Clempner, 2005, 2006). The GPN structure will represent all possible strategies existing within the game.

Definition 2. A *game Petri net* is the 8-tuple $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$, where

- $\mathcal{N} = \{1, 2, \dots, n\}$ denotes a finite set of players,
- $P = P_1 \times P_2 \times \cdots \times P_n$ is the set of places that represents the Cartesian product of states (each tuple is represented by a place),
- $Q = Q_1 \times Q_2 \times \cdots \times Q_n$ is the set of transitions that represents the Cartesian product of the conditions (each tuple is represented by a transition),
- $F \subseteq I \cup O$ is a set of arcs where $I \subseteq (P \times Q)$ and $O \subseteq (Q \times P)$ such that $P \cap Q = \varnothing$ and $P \cup Q \neq \varnothing$,
- $W : F \to N^n$ is a weight function,
- M_0: $P \to N^n$ is the initial marking,
- $\pi : I \to \mathbb{R}^n_+$ is a routing policy representing the probability of choosing a particular transition (routing arc), such that for each $p_i \in P$

$$\sum_{q_{j\iota} : (p_{i\iota}, q_{j\iota}) \in I} \pi((p_{i\iota}, q_{j\iota})) = 1, \quad \forall \iota \in \mathcal{N},$$

- $U : P \to \mathbb{R}^n_+$ is a trajectory function.

Interpretation: The previous behavior of the *GPN* is described as follows: When a token reaches a place, it is reserved for the firing of a given transition according to the routing policy determined by U. A transition q must fire as soon as all places $p_1 \in P$ contain enough tokens reserved for the transition q. Once the transition fires, it consumes the corresponding tokens and immediately produces an amount of tokens in each subsequent place $p_2 \in P$. The satisfaction of $\pi(\delta) = 0$ for $\delta \in I$ means that there are no arcs in the place-transitions Petri net.

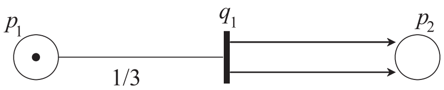

Fig. 1. Routing policy, Case 1.

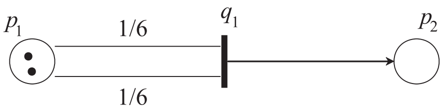

Fig. 2. Routing policy, Case 2.

In Figs. 1 and 2 partial routing policies π for a given player $\iota \in \mathcal{N}$ are represented with respect to a game Petri net GPN that generates a transition from the state p_1 to the state p_2 where $p_1, p_2 \in P$:

- *Case 1.* In Fig. 1 the probability that q_1 generates a transition from the state p_1 to p_2 is 1/3. But, because the q_1 transition to the state p_2 has two arcs, the probability to generate a transition from the state p_1 to p_2 is increased to 2/3. Note that one token is required in p_1 to fire the transition q_1, and two tokens are generated in p_2.
- *Case 2.* In Fig. 2 we set by convention the probability that q_1 generates a transition from the state p_1 to p_2 as 1/3 (1/6 plus 1/6). However, because the transition q_1 to state p_2 has only one arc, the probability to generate a transition from state p_1 to p_2 is decreased to 1/6. Note that two tokens are required in p_1 to fire the transition q_1, and one token is generated in p_2.
- *Case 3.* Finally, we have the trivial case when there exists only one arc from p_1 to q_1 and from q_1 to p_2.

Remark 1. The previous definition in no way changes the behavior of the place-transition Petri net, and the *routing policy is used to calculated the trajectory value at each place of the net.*

Remark 2. It is important to note that the trajectory value can be re-normalized after each transition or time k of the net.

Remark 3. For the case of n players, we will represent the routing policies π using the Cartesian product, i.e., (1/3, 1/5, 1/16, ...).

It is important to note that, by definition, the trajectory function U is employed only for establishing trajectory tracking, working in a different execution level of that of the place-transition Petri net. The trajectory function U changes in no way the place-transition Petri net's evolution or performance.

$U_k(\cdot)$ denotes the trajectory value at the place $p_i \in P$ at the time k. Let $[U_k] = [U_k(\cdot), \ldots, U_k(\cdot)]^T$ denote the trajectory value state of the GPN at the time k. $FN : F \to \mathbb{R}_+$ is the number of arcs from place p to the transition q (the number of arcs from the transition q to the place p). The rest of GPN functionality is as described in PN preliminaries.

Let us recall some basic notions in game theory. We denote by $S_\iota = \{s_i\}$ the set of pure strategies for the player ι (strategies are represented by the probability that a transition can be fired in the GPN). For notational convenience, we write $S = \prod_{\iota \in \mathcal{N}} S_\iota$ (the pure strategies profile), and $S_{-\iota} = \prod_{j \in \mathcal{N}|\{\iota\}} S_j$ (the pure strategies profile of all the players but for the player ι). For an action tuple $s = (s_1, \ldots, s_n) \in S$ we write $s_{-\iota} = (s_1, \ldots, s_{\iota-1}, s_{\iota+1}, \ldots, s_n)$ and, with an abuse of notation, $s = (s_\iota, s_{-\iota})$.

Similarly, we denote by $\Gamma_\iota = \{\sigma_i\}$ the set of mixed strategies for the player ι, identified with the routing policy representing the probability of choosing a particular transition. Analogously, we use $\Gamma = \prod_{\iota \in \mathcal{N}} \Gamma_\iota$ to denote the mixed strategies profile that combines strategies, one for each player, and $\Gamma_{-\iota} = \prod_{j \in \mathcal{N}|\{\iota\}} \Gamma_j$ to denote the mixed strategies profile of all the players except for the player ι. For a strategy tuple $\sigma = (\sigma_1, \ldots, \sigma_n) \in \Gamma$ we write $\sigma_{-\iota} = (\sigma_1, \ldots, \sigma_{\iota-1}, \sigma_{\iota+1}, \ldots, \sigma_n)$ and, with an abuse of notation, $\sigma = (\sigma_\iota, \sigma_{-\iota})$. For a strategy profile $\sigma_{-\iota}$, we write $\sigma_{-\iota} = \prod_{j \in \mathcal{N}|\{\iota\}} \sigma_j$, the probability identified with the routing policy π that the opponents of the player ι play the strategy profile $s_{-i} \in S_{-i}$. We restrict our attention to independent strategy profiles. For our construction of the GPN, a strategy profile determines an outcome representing the corresponding trajectory value of each player.

Then, formally, we introduce the following definitions:

Definition 3. A *final decision point* $p_f \in P$ with respect to a game Petri net $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ is a place $p \in P$ where the infimum is asymptotically approached (or the minimum is attained), i.e., $U(p) = \bar{\mathbf{0}}$ or $U(p) = \bar{\mathbf{C}}$.

Definition 4. An *optimum point* $p^{\triangle} \in P$ with respect a game Petri net $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ is a final decision point $p_f \in P$ where the best choice is selected 'according to some criteria'.

Property 1. *Every game Petri net* $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ *has a final decision point.*

Definition 5. A *strategy with respect a game Petri net* $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ is identified by σ and consists of the routing policy transition sequence represented in the GPN graph model such that some point $p \in P$ is reached.

Definition 6. An *optimum strategy with respect to a game Petri net* $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ is identified by $\sigma^{\triangle}$ and consists of the routing policy transition sequence represented in the GPN graph model such that an optimum point $p^{\triangle} \in P$ is reached.

Remark 4. It is important to note that a strategy can be conceptualized in a different manner depending on the implementation point of view. It can be implemented as the probability that a transition can be fired, as usual, or, more generally, as a chain of such probabilities. Both perspectives are correct, but in the latter case we only have to give an interpretation of strategy optimality in terms of the chain of transitions.

Consider arbitrary $p_i \in P$ and for each fixed transition $q_j \in Q$ that forms an output arc $(q_j, p_i) \in O$. We look at all previous places p_h of the place p_i denoted by the list (set) $p_{\eta_{ij}} = \{p_h : h \in \eta_{ij}\}$, where $\eta_{ij} = \{h : (p_h, q_j) \in I \quad \& \quad (q_j, p_i) \in O\}$, which materialize all input arcs $(p_h, q_j) \in I$ and form the sum

$$\sum_{h \in \eta_{ij}} \left(\left\langle \sigma_{hj}(p_i) * U_k^{\sigma_{hj}}(p_h) \right\rangle\right)_{\iota} \tag{4}$$

where

$$\sigma_{hj}(p_i) = \Bigg(\pi(p_{h_1}, q_{j_1}) * \frac{FN(q_j, p_i)}{FN(p_h, q_j)}, \pi(p_{h_2}, q_{j_2}) * \frac{FN(q_j, p_i)}{FN(p_h, q_j)}, \dots, \pi(p_{h_n}, q_{j_n}) * \frac{FN(q_j, p_i)}{FN(p_h, q_j)}\Bigg),$$

$(\langle * \rangle)_{\iota}$ representing the product of the vector element by element, i.e., $(\langle (a_1, a_2, \dots, a_n) * (b_1, b_2, \dots, b_n) \rangle)_{\iota} = (a_1 b_1, a_2 b_2, \dots, a_n b_n)$. p_{h_ι} is the ι-th element of the tuple routing policy π, and the index sequence j_{ι} is the set $\{j_\iota : \forall \iota \; q_{j_\iota} \in (p_{h_\iota}, q_{j_\iota}) \cap (q_{j_\iota}, p_{i_\iota})$ & p_{h_ι} running over the set $p_{\eta_{ij}}\}$. The quotient $FN(q_j, p_i)/FN(p_h, q_j)$ is used for normalizing the routing policies π. Note that in the formula of $\sigma_{hj}(p_i)$ it is not necessary to specify $\forall \iota \; : \; FN(q_{j_\iota}, p_{i_\iota})$ and $FN(p_{h_\iota}, q_{j_\iota})$ for calculating $FN(q_{j_\iota}, p_{i_\iota})/FN(p_{h_\iota}, q_{j_\iota})$ because the number of arcs $(FN(\cdot, \cdot))$ is the same for all players.

Proceeding with all q_js for a given player $\iota \in \mathcal{N}$, we form the vector indexed by the sequence j identified by $(j_0, j_1, \dots, j_f)$ as follows:

$$\alpha = \left[\; \alpha_{j_0}, \dots, \alpha_{j_f} \; \right], \tag{5}$$

where

$$\alpha_{j_\ell} = \sum_{h \in \eta_{i j_\ell}} \left(\left\langle \sigma_{h j_\ell}(p_i) * U_k^{\sigma_{h j_\ell}}(p_h) \right\rangle\right)_{\iota}.$$

Intuitively, the vector (5) represents all possible trajectories through the transitions q_j to a place p_i for a fixed i and a given player $\iota \in \mathcal{N}$.

Continuing the construction of the definition of the trajectory function U, let us introduce the following definition:

Definition 7. Let $L : \mathbb{R}^n \to \mathbb{R}_+$ be a continuous map and let x be the state of a realized trajectory. Then, L is a *vector Lyapunov-like function* (Kalman and Bertram, 1960; Lakshmikantham and Martynyuk, 1990; Lakshmikantham *et al.*, 1991) if it satisfies the following properties:

1. $L(x_1, \dots, x_n) = \Big[L_1(x_1), \quad \dots, \quad L_n(x_n) \Big]$,
2. if $\exists x_i^*$ such that $\forall i \; L_i(x_i^*) = 0$, then $L(x_1^*, \dots, x_n^*) = \bar{\mathbf{0}}$,
3. if $L_i(x_i) > 0$ for $\forall x_i \neq x_i^*$ then $L(x_1, \dots, x_n) > \bar{\mathbf{0}}$,
4. if $L_i(x_i) \to \infty$ when $x_i \to \infty$, then $L(x_1, \dots, x_n) \to \infty$,
5. if $\Delta L_i = L_i(x_i) - L_i(y_i) < 0$ for all $(y_1, \dots, y_n) \leq_U (x_1, \dots, x_n)$ and $(x_1, \dots, x_n)$, $(y_1, \dots, y_n) \neq (x_1^*, \dots, x_n^*)$, then $\Delta L = L(x_1, \dots, x_n) - L(y_1, \dots, y_n) < \bar{\mathbf{0}}$.

From the previous definition we have the following remark:

Remark 5. In Definition 7, Point 4 we state that if $L_i(x_i) \to \infty$ when $x_i \to \infty$, then $L(x_1, \dots, x_n) \to \infty$, meaning that there is no x^* reachable from some x.

Then, formally we define the trajectory function U as follows:

Definition 8. The *trajectory function* U for a given player $\iota \in \mathcal{N}$ with respect a game Petri net $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ is represented as

$$U_{k,\iota}^{\sigma_{hj}}(p_i) = \begin{cases} U_k(p_0) & \text{if} \quad i = 0, \; k = 0, \\ L(\alpha) & \text{if} \quad i > 0, \; k = 0 \\ & \text{or} \quad i \geq 0, \; k > 0, \end{cases} \tag{6}$$

where the vector function $L : D \subseteq \mathbb{R}_+^n \to \mathbb{R}_+^n$ is a vector Lyapunov-like function which optimizes the trajectory value through all possible strategies (i.e., through all possible trajectories defined by the different q_js), D is the decision set formed by the js $(0 \leq j \leq f)$ of all those possible transitions $(q_j, p_i) \in O$, and α is given in (5).

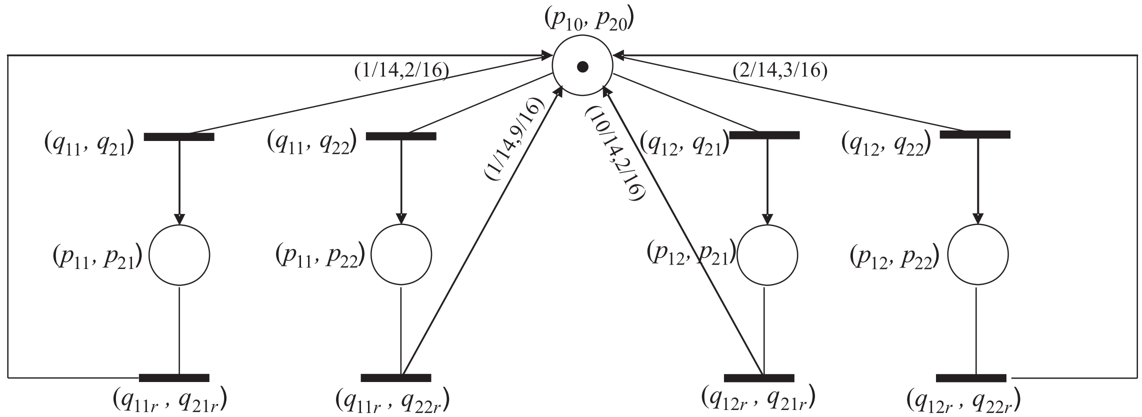

Fig. 3. GPN — iterated prisoner's dilemma.

Remark 6. From the previous definition we have the following remarks:

- The vector Lyapunov-like function L associated with Definition 4 regarding an optimum point guarantees that the optimal course of action is followed. In addition, the vector function L establishes a preference relation because, by definition, L is asymptotic and the criteria established in Definition 4 give the decision maker an opportunity to select a path that optimizes the trajectory value.
- The iteration over k for U is as follows:
  1. for $i = 0$ and $k = 0$ the trajectory value is $U_0(p_0)$ at place p_0 and for the rest of the places p_i the value is $\bar{\mathbf{0}}$,
  2. for $i \geq 0$ and $k > 0$ the trajectory value is $U_k^{\sigma_{hj}}(p_i)$ at each place p_i, and it is computed by taking into account the value of the previous places p_h for k and $k-1$ (when needed).

The *prisoner's dilemma* (Axelrod, 1984) is used as a first approach in game theory to conceptualize the conflict between mutual support and selfish exploitation among interacting players. The game can be illustrated by an example where two men are arrested for a crime. The police tell each suspect separately that if he testifies against the other, he will be rewarded for cooperating. Each prisoner has two possible strategies (Table 1): to testify (cooperate) or to defect with the police (not to testify). If no player cooperates, there is a mutual punishment with a score of P (the punishment corresponding to mutual defection, in this particular case equal to zero, given that there is supposedly no proof to convict either of the two). If both testify, there is a mutual reduction in punishment, resulting in a penalty value of R. However, if one testifies and the other does not, the testifier receives a considerable punishment reduction (the penalty of T, the temptation for defection), and the other player receives the regular punishment (the penalty of S, the "sucker" penalty for attempting to cooperate against defection). This game has usually two equilibrium points: one non-cooperative (neither of the prisoners testifying) and the other cooperative (both prisoners testify to the police).

Table 1. Prisoner's dilemma

Player 1 \ Player 2	**Cooperate (testify)**	**Defect (not testify)**
Cooperate (testify)	*R*, **R**	*S*, **T**
Defect *(not testify)*	*T*, **S**	*P*, **P**

Let us suppose that $T > R > P > S$. It is easy to see that we have the structure of a dilemma like the one in the story. On the one had, let us suppose that Player 2 testifies. Then Player 1 obtains R for cooperating and T for defecting, and so the is better off defecting. On the other hand, let us suppose that Player 2 does not testify. Then Player 1 obtains S for cooperating and P for defecting, and so he is again better off defecting. The move 'not to testify' for Player 1 is said to strictly dominate the move 'testify': whatever his opponent does, he is better off choosing 'not testify' than 'testify'. By symmetry, 'not to testify' also strictly dominates 'testify' for Player 2. Thus, two "rational" players will defect and receive a payoff of P, while two "irrational" players can cooperate and receive a greater payoff R.

Example 1. The Iterated Prisoner's Dilemma (IPD), represented by Fig. 3, is played in the same manner as the classical prisoner's dilemma, but assumes that the players will interact with each other more than once. Axelrod (1984) demonstrates that strategies that allow for cooperation will usually have higher scores than strategies of pure non-cooperation. A player applying the Tit-for-Tat (TFT) strategy will cooperate at the beginning, and when he is exploited, will return to the last action of his/her opponent.

For the prisoner's dilemma, a Lyapunov equilibrium point is particularly interesting when the players have no motivation to unilaterally deviate from it and improve their outcome (as the Nash property requires) nor cooperation. In this sense, the Lyapunov equilibrium point improves the Nash equilibrium point to the fact that it is resistant to cooperation deviations between players. This is because the stability achieved by the Nash equilibrium is defined to avoid only unilateral deviations of each player. The best-response against TFT with respect to the trajectory function implemented as a Lyapunov-like function $U_\iota^{(\sigma_\iota,\sigma_{-\iota})}(p)$ *is always 'not to cooperate'.*

A strategy σ for a time $k=0,\ i\geq 0$ is

$$\begin{bmatrix} 1 & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(11,21)}((p_{11},p_{21})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(11,22)}((p_{11},p_{22})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(12,21)}((p_{12},p_{21})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(12,22)}((p_{12},p_{22})) & 0\ 0\ 0\ 0 \end{bmatrix}\begin{bmatrix} U((p_{10},p_{20})) \\ U((p_{11},p_{21})) \\ U((p_{11},p_{22})) \\ U((p_{12},p_{21})) \\ U((p_{12},p_{22})) \end{bmatrix}.$$

A strategy $\sigma^{'}$ leading to the cooperative equilibrium in the place (p_{12},p_{22}) for a time $k=1,\ i\geq 0$ is

$$\begin{bmatrix} 1 & 0\ 0\ 0\ B \\ \sigma_{(10,20),(11,21)}((p_{11},p_{21})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(11,22)}((p_{11},p_{22})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(12,21)}((p_{12},p_{21})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(12,22)}((p_{12},p_{22})) & 0\ 0\ 0\ 0 \end{bmatrix}\begin{bmatrix} U((p_{10},p_{20})) \\ U((p_{11},p_{21})) \\ U((p_{11},p_{22})) \\ U((p_{12},p_{21})) \\ U((p_{12},p_{22})) \end{bmatrix},$$

where $B=\sigma_{(12,21),(12r,22r)}((p_{10},p_{20}))$.

A strategy $\sigma^{''}$ leading to the non-cooperative equilibrium in the place (p_{11},p_{21}) for a time $k\geq 2,\ i\geq 0$ is

$$\begin{bmatrix} 1 & D\ 0\ 0\ 0 \\ \sigma_{(10,20),(11,21)}((p_{11},p_{21})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(11,22)}((p_{11},p_{22})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(12,21)}((p_{12},p_{21})) & 0\ 0\ 0\ 0 \\ \sigma_{(10,20),(12,22)}((p_{12},p_{22})) & 0\ 0\ 0\ 0 \end{bmatrix}\begin{bmatrix} U((p_{10},p_{20})) \\ U((p_{11},p_{21})) \\ U((p_{11},p_{22})) \\ U((p_{12},p_{21})) \\ U((p_{12},p_{22})) \end{bmatrix},$$

where $D=\sigma_{(11,21),(11r,21r)}((p_{10},p_{20}))$. ♦

In most cases there is more than one best-response strategy. For example, cooperation with TFT can be achieved by the strategy 'always cooperate', by another TFT or by many other cooperative strategies. Notice that the strategies in the IPD will depend on the two premises (a rational belief about other players' strategies, the correctness of belief) under which the game is being played.

Notation 2. With the intention to further facilitate the notation, we will represent the trajectory function U as follows:

1. $U_k(p_i)\triangleq U_k^{\sigma_{hj}}(p_i)$ for any transition and any strategy,
2. $U_k^{\triangle}(p_i)\triangleq U_k^{\sigma_{hj}^{\triangle}}(p_i)$ for an optimum transition and an optimum strategy,
3. σ_{hj} by σ, and we will specify h_j when it is necessary to describe the trajectory of the strategy σ in the GPN,
4. $U_{k,\iota}$ by U_ι representing the trajectory function of player ι, and we will specify the index k denoting the trajectory value as U_k when it is needed.

The reader will easily identify which notation is used depending on the context.

Property 2. *The continuous function $U(\cdot)$ satisfies, moreover, the following properties:*

1. *There exists $p^{\triangle}\in P$ such that*
   - (a) *if there exists an infinite sequence $\{p_i\}_{i=1}^{\infty}\in P$ with $p_n \underset{n\to\infty}{\to} p^{\triangle}$ such that $\bar{\mathbf{0}}\leq\cdots<U(p_n)<U(p_{n-1})<\cdots<U(p_1)$, then $U(p^{\triangle})$ is the infimum, i.e., $U(p^{\triangle})=\bar{\mathbf{0}}$,*
   - (b) *if there exists a finite sequence $p_1,\dots,p_n\in P$ with $p_1,\dots,p_n\to p^{\triangle}$ such that $\bar{\mathbf{C}}=U(p_n)<U(p_{n-1})<\cdots<U(p_1)$, then $U(p^{\triangle})$ is the minimum, i.e., $U(p^{\triangle})=\bar{\mathbf{C}}$, where $\bar{\mathbf{C}}\in\mathbb{R}^n,(p^{\triangle}=p_n)$,*
2. *$\max\{U(p)>\bar{\mathbf{0}},U(p)>\bar{\mathbf{C}}\}$ where $\bar{\mathbf{C}}\in\mathbb{R}^n$, $\forall p\in P$ such that $p\neq p^{\triangle}$,*
3. *$\forall p_i,p_{i-1}\in P$ such that $p_{i-1}\leq_U p_i$. Then $\Delta U=U(p_i)-U(p_{i-1})<\bar{\mathbf{0}}$.*

From the previous property we have the following remark:

Remark 7. In Property 2, Point 3 we state that $\Delta U=U(p_i)-U(p_{i-1})<\bar{\mathbf{0}}$ for determining the asymptotic condition of the Lyapunov-like function.

Property 3. *The trajectory function $U:P\to\mathbb{R}_+^n$ is a vector Lyapunov-like function.*

Remark 8. From Properties 2 and 3 we have that

- $U(p^{\triangle})=\bar{\mathbf{0}}$ or $U(p^{\triangle})=\bar{\mathbf{C}}$ means that a final state is reached. Without lost generality we can say that $U(p^{\triangle})=\bar{\mathbf{0}}$ by means of a translation to the origin.
- In Property 2 we determine that the vector Lyapunov-like function $U(p)$ approaches a infimum/minimum when p is large thanks to Point 4 of Definition 6.

- In Property 2, Point 3 is equivalent to the following statement: $\exists\{\bar{\varepsilon}_i\}$, $\bar{\varepsilon}_i > \bar{\mathbf{0}}$ such that $|U(p_i) - U(p_{i-1})| > \bar{\varepsilon}_i$, where $\forall p_i, p_{i-1} \in P$ we have that $p_{i-1} \leq_U p_i$.

Explanation. Intuitively, a Lyapunov-like function can be considered as a routing function and an optimal cost function. In our case, an optimal discrete problem, the cost-to-target values are calculated using a discrete Lyapunov-like function. Every time a discrete vector field of possible actions is calculated over the decision process. Each applied optimal action (selected via some 'criteria') decreases the optimal value, ensuring that the optimal course of action is followed and establishing a preference relation. In this sense, the criteria change the asymptotic behavior of the Lyapunov-like function by an optimal trajectory tracking value. It is important to note that the process finishes when the equilibrium point is reached. This point determines a significant difference with Bellman's equation.

Definition 9. Let $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ be a game Petri net. A *trajectory* ω is an (finite or infinite) ordered subsequence of places $p_{\varsigma(1)} \leq_{U_k} p_{\varsigma(2)} \leq_{U_k} \cdots \leq_{U_k} p_{\varsigma(n)} \leq_{U_k} \cdots$ such that a given strategy σ holds.

Definition 10. Let $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ be a game Petri net. An *optimum trajectory* ω is an (finite or infinite) ordered subsequence of places $p_{\varsigma(1)} \leq_{U_k^\triangle} p_{\varsigma(2)} \leq_{U_k^\triangle} \cdots \leq_{U_k^\triangle} p_{\varsigma(n)} \leq_{U_k^\triangle} \cdots$ such that the optimum strategy $\sigma^\triangle$ holds.

Theorem 1. *Let $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ be a non-blocking game Petri net (unless $p \in P$ is an equilibrium point). Then we have*

$$U_k^\triangle(p^\triangle) \leq U_k(p), \quad \forall \sigma, \sigma^\triangle.$$

Proof. Let U be defined as in (6). Then, starting from p_0 and proceeding with the iteration, eventually the trajectory ω given by $p_0 = p_{\varsigma(1)} \leq_{U_k} p_{\varsigma(2)} \leq_{U_k} \cdots \leq_{U_k} p_{\varsigma(n)} \leq_{U_k} \cdots$ which converges to $p^\triangle$, i.e., the optimum trajectory is obtained. Since at the optimum trajectory the optimum strategy $\sigma^\triangle$ holds, we have that $U_k^\triangle(p^\triangle) \leq U_k(p)$, $\forall \sigma, \sigma^\triangle$. ■

Remark 9. The inequality $U_k^\triangle(p^\triangle) \leq U_k(p)$ means that the trajectory value is optimum when the optimum strategy is applied.

Corollary 1. *Let $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ be a non-blocking game Petri net (unless $p \in P$ is an equilibrium point), and let $\sigma^\triangle$ be an optimum strategy. Set* $L = \left[\min_{i=1,\dots,|\alpha|}\{\alpha_i\}, \quad \dots, \quad \min_{i=1,\dots,|\alpha|}\{\alpha_i\}\right]$. *Then*

$$U_k^\triangle(p) = \underbrace{\begin{bmatrix} \sigma_{0j_m}^\triangle(p_{\varsigma(0)}) & \sigma_{1j_m}^\triangle(p_{\varsigma(0)}) & \cdots & \sigma_{nj_m}^\triangle(p_{\varsigma(0)}) \\ \sigma_{0j_n}^\triangle(p_{\varsigma(1)}) & \sigma_{1j_n}^\triangle(p_{\varsigma(1)}) & \cdots & \sigma_{nj_n}^\triangle(p_{\varsigma(1)}) \\ \cdots & \cdots & \cdots & \cdots \\ \sigma_{0j_v}^\triangle(p_{\varsigma(i)}) & \sigma_{1j_v}^\triangle(p_{\varsigma(i)}) & \cdots & \sigma_{nj_v}^\triangle(p_{\varsigma(i)}) \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix}}_{\sigma^\triangle} \underbrace{\begin{bmatrix} U_k(p_0) \\ U_k(p_1) \\ \cdots \\ U_k(p_i) \\ \cdots \end{bmatrix}}_{U}, \tag{7}$$

where p is a vector whose elements are those places which belong to the optimum trajectory ω given by $p_0 \leq p_{\varsigma(1)} \leq_{U_k} p_{\varsigma(2)} \leq_{U_k} \cdots \leq_{U_k} p_{\varsigma(n)} \leq_{U_k} \cdots$ which converges to $p^\triangle$.

Proof. Since at each step of the iteration, $U_k^\triangle(p_i)$ is equal to one of the elements of vector α, we have that the representation that describes the dynamical trajectory behavior of tracking the optimum strategy $\sigma^\triangle$ is given in (7), where $j_m, j_n, \dots, j_v, \dots$ represent the indexes of the optimal routing policy, defined by the q_js. ■

4. Lyapunov-Nash Equilibrium Point

The interaction among players obliges each player to develop a belief about the possible strategies of the other players. Nash equilibria (Nash, 1951, 1996, 2002) are supported by two premises: (i) each player behaves rationally given the beliefs about the other players' strategies; and (ii) these beliefs are correct. Both premises allow us to regard the Nash equilibrium point as a steady state of the strategic interaction. In particular, the second premise makes this an equilibrium concept, because when every individual is acting in agreement with the Nash equilibrium, no one has the need to take another strategy.

The best-reply strategy for a player is relative to the strategy profile chosen by the opponents. The strategy profile is said to contain a best reply for a given player if he or she cannot increase the utility by playing another strategy with respect to the opponents' strategies. A strategy profile is a Nash equilibrium point if none of the players can increase the utility by playing another strategy. In other words, each player's choice of a strategy is a best reply to the strategies taken by his opponents. This is when a player acting in accordance with the Nash equilibrium has no motivation to unilaterally deviate and take another strategy. Formally, we have the following definitions:

Consider the game $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$. For each player $\iota \in \mathcal{N}$ and each profile $\sigma_{-\iota} \in \Gamma_{-\iota}$ of the strategies of his opponent, introduce the set of *best* *replies*, i.e., the strategies that player ι cannot improve. It is defined as follows:

$$B_\iota(\sigma_{-\iota}) := \Big\{\sigma_\iota^\triangle \in \Gamma_\iota | \forall \sigma'_\iota \in \Gamma_\iota : U_\iota^{(\sigma_\iota^\triangle, \sigma_{-\iota})}(p^\triangle) \leq U_\iota^{(\sigma'_\iota, \sigma_{-\iota})}(p)\Big\}. \tag{8}$$

Remark 10. In contrast to what we define in game theory, in a GPN, by the definition of the Lyapunov-like trajectory function, we look for an equilibrium point at the minimum, for that reason we change '$\geq$' to '$\leq$' in (8). Since Γ_ι is finite and u_ι establishes an acyclic order, $B_\iota(\sigma_{-\iota})$ is not empty.

Remark 11. It is important to note that in case the strategy is implemented as a chain of transitions, '$\leq$' does not represent a vector inequality, and the interpretation is obtained from calculating the best reply B_ι.

A Nash equilibrium is a profile of strategies such that each player's strategy is an optimal response to the other players' strategies.

Definition 11. A strategy profile $\sigma_\iota^\triangle$ is a *Nash equilibrium point* if, for all players ι,

$$U_\iota^{(\sigma_\iota^\triangle, \sigma_{-\iota}^\triangle)}(p^\triangle) \leq U_\iota^{(\sigma'_\iota, \sigma_{-\iota}^\triangle)}(p) \forall \sigma'_\iota \in \Gamma_\iota. \tag{9}$$

Note that in (9) we use '$\leq$' instead of '$\geq$' by the arguments established in Remark 10.

Remark 12. Intuitively, a Nash equilibrium is a strategy profile for a game, such that no player can increase his or her payoff by changing the strategy, while the other players keep their strategies fixed.

Definition 12. A strategy σ has the *fixed point property* if it leads to the optimum point $U_\iota^{(\sigma_\iota^\triangle, \sigma_{-\iota}^\triangle)}(p^\triangle)$.

Remark 13. From the previous two definitions, the following characterization is obtained: A strategy which has the fixed point property is equivalent to being a Nash equilibrium point.

Theorem 2. *A non-blocking (unless $p \in P$ is an equilibrium point) game Petri net $GPN = (\mathcal{N}, P, Q, F, W, M_0, \pi, U)$ has a strategy σ which has the fixed point property.*

Proof. The conclusion is a direct consequence of Theorem 1 and its proof (where the existence of $p^\triangle$ is guaranteed by the first property given in the definition of the Lyapunov-like function, given by Definition 7). ■

Corollary 2. *If, in addition to the hypothesis of Theorem 2, the game GPN is finite, the strategy σ leads to an equilibrium point.*

Proof. See Corollary 1. ■

Theorem 3. *The optimum point*[1] *coincides with Nash equilibria.*

Proof. This is immediate from Definition 4 of an optimum point and Remark 13. ■

Remark 14. The potential of the previous theorem consists in the simplicity of its formal proof regarding the existence of an equilibrium point (in contrast to the fact that a Nash equilibrium point exists in a non-empty, compact, convex subset of a Euclidian space by Kakutani's fixed-point theorem).

5. Conclusions and Future Work

A formal framework for the shortest path game representation called game Petri nets was presented. The expressive power and the mathematical formality of the GPN contribute to bridging the gap between Petri nets and game theory. The Lyapunov method induces a new equilibrium and stability concept in game theory. We proved that the equilibrium concept in a Lyapunov sense coincides with the equilibrium concept of Nash, representing an alternative way to calculate the equilibrium of the game. As far as we know, we introduce the game theory as a new application area in Petri net theory. Moreover, we introduce a new type of equilibrium point in the Lyapunov sense to game theory, lending to the game necessary and sufficient conditions of stability (Clempner, 2005) under certain restrictions. As for future work, we will develop factors affecting the equilibria and their relationship with Nash, Pareto and Lyapunov equilibrium points.

References

Axelrod R. (1984): *The Evolution of Cooperation*. — New York: Basic Books.

Bellman R.E. (1957): *Dynamic Programming*. — Princeton: Princeton University Press.

Bertsekas D.P. and Shreve S.E. (1978): *Stochastic Optimal Control: The Discrete Time Case*. — New York: Academic Press.

Bertsekas D.P. (1987): *Dynamic Programming: Deterministic and Stochastic Models*. —- Englewood Cliffs: Prentice–Hall.

Bertsekas D.P. and Tsitsiklis J.N. (1989): *Parallel and Distributed Computation: Numerical Methods*. — Englewood Cliffs: Prentice–Hall.

[1] The definition of an optimum point is equivalent to the definition of a "steady state" equilibrium point in the Lyapunov sense given by Kalman (1960).

Bertsekas D.P. and Tsitsiklis J.N. (1991): *An analysis of stochastic shortest path problems*. — Math. Oper. Res., Vol. 16, No. 3, pp. 580–595.

Blackwell D. (1967): *Positive dynamic programming*. — Proc. 5th Berkeley Symp. *Math., Statist., and Probability*, Berkeley, California, Vol. 1, pp. 415–418.

Clempner J. (2005): *Colored decision process petri nets: modeling, analysis and stability*. — Int. J. Appl. Math. Comput. Sci., Vol. 15, No. 3, pp. 405-420.

Clempner J. (2006): *Towards modeling the shortest path problem and games with petri nets*. — Proc. Doctoral Consortium at 27-th Int. Conf. *Application and Theory of Petri Nets and Other Models Of Concurrency*, Turku, Finland, pp. 1–12.

Derman C. (1970): *Finite State Markovian Decision Processes*. — New York: Academic Press.

Dynkin E.B. (1963): *The optimum choice of the instant for stopping a Markov process*. — Soviet Math. Doklady, Vol. 150, pp. 238–240.

Eaton J.H. and Zadeh L.A. (1962): *Optimal pursuit strategies in discrete state probabilistic systems*. — Trans. ASME Ser. D, J. Basic Eng., Vol. 84, pp. 23–29.

Grigelionis R.I. and Shiryaev A.N. (1966): *On Stefan's problem and optimal stopping rules for Markov processes*. — Theory Prob. Applic., Vol. 11, pp. 541–558.

Hernández-Lerma O. and Lasserre J.B. (1996): *Discrete-Time Markov Control Process: Basic Optimality Criteria*. — Berlin: Springer.

Hernández-Lerma O., Carrasco G. and Pére-Hernández R. (1999): *Markov Control Processes with the Expected Total Cost Criterion: Optimality*. — Stability and Transient Model. Acta Applicadae Matematicae, Vol. 59, No. 3, pp. 229–269.

Hernández-Lerma O. and Lasserre J.B. (1999): *Futher Topics on Discrete-Time Markov Control Process*. — Berlin: Springer.

Hinderer K. and Waldmann K.H (2003): *The critical discount factor for finite Markovian decision process with an absorbing set*. — Math. Meth. Oper. Res., Vol. 57, pp. 1–19.

Hinderer K. and Waldmann K.H. (2005): *Algorithms for countable state Markov decision model with an absorbing set*. — SIAM J. Contr. Optim., Vol. 43, pp. 2109–2131.

Howard R.A. (1960): *Dynamic Programming and Markov Processes*. — Cambridge, MA: MIT Press.

Kalman R.E. and Bertram J.E. (1960): *Control system analysis and design via the "Second Method" of Lyapunov*. —- J. Basic Eng., ASME, Vol. 82(D), pp. 371–393.

Kuhn H.W. and Nasae S. (Eds.) (2002): *The Essential John Nash*. — Princeton: Princeton University Press.

Kumar P.R. and Shiau T.H. (1981): *Zero sum dynamic games*, In: Control and Dynamic Games, (C.T. Leondes, Ed.) — Academic Press, pp. 1345–1378.

Kushner H.J. and Chamberlain S.G. (1969): *Finite state stochastic games: Existence theorems and computational procedures*. — IEEE Trans. Automat. Contr., Vol. 14, No. 3.

Kushner H. (1971): *Introduction to Stochastic Control*. — New York: Holt, Rinehart and Winston.

Lakshmikantham S. Leela and Martynyuk A.A. (1990): *Practical Stability of Nonlinear Systems*. — Singapore: World Scientific.

Lakshmikantham V., Matrosov V.M. and Sivasundaram S. (1991): *Vector Lyapunov Functions and Stability Analysis of Nonlinear Systems*. — Dordrecht: Kluwer.

Mandl P. and Seneta E. (1969): *The theory of non-negative matrices in a dynamic programming problem*. — Austral. J. Stat., Vol. 11, pp. 85–96.

Nash J. (1951): *Non-cooperative games*. — Ann. Math., Vol. 54, pp. 287–295.

Nash J. (1996): *Essays on Game Theory*. — Cheltenham: Elgar.

Pallu de la Barriere R. (1967): *Optimal Control Theory*. — Philadelphia: Saunders.

Patek S.D. (1997): *Stochastic Shortest Path Games: Theory and Algorithms*. — Ph.D. thesis, Dep. Electr. Eng. Comput. Sci., MIT, Cambridge, MA.

Patek S.D. and Bertsekas D.P. (1999): *Stochastic shortest path games*. — SIAM J. Contr. Optim., Vol. 37, No. 3, pp. 804–824.

Patek S.D. (2001): *On terminating Markov decision processes with a risk averse objective function*. — Automatica, Vol. 37, No. 9, pp. 1379–1386.

Pliska S.R. (1978): *On the transient case for Markov decision chains with general state space*, In: Dynamic Programming and Its Applications, (M.L. Puterman, Ed.). — New York: Springer, pp 335–349.

Puterman M.L. (1994): *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. — New York: Wiley.

Rieder U. (1975): *Bayesian dynamic programming*. — Adv. Appl. Prob., Vol. 7, pp. 330–348.

Shapley L.S. (1953): *Stochastic Games*. — Proc. National. Acad. Sci., Mathematics, Vol. 39, pp. 1095–1100.

Shiryaev A.N. (1978): *Optimal Stopping Problems*. — New York: Springer.

Strauch R. (1966): *Negative dynamic programming*. — Ann. Math. Stat., Vol. 37, pp. 871–890.

Van der Wal J. (1981): *Stochastic dynamic programming*. — Math. Centre Tracts 139, Mathematisch Centrum, Amsterdam.

Veinott A.F., Jr. (1969): *Discrete dynamic programming with sensitive discount optimality criteria*. — Ann. Math. Stat., Vol. 40, No. 5, pp. 1635–1660.

Whittle P. (1983): *Optimization over Time*. — New York: Wiley, Vol. 2.

Received: 29 October 2005
Revised: 12 April 2006
Re-revised: 7 July 2006