Pure stationary optimal strategies in Markov decision processes

Hugo Gimbert

LIX, Ecole Polytechnique, France *
hugo.gimbert@laposte.net

Abstract. Markov decision processes (MDPs) are controllable discrete event systems with stochastic transitions. Performances of an MDP are evaluated by a payoff function. The controller of the MDP seeks to optimize those performances, using optimal strategies.
There exists various ways of measuring performances, i.e. various classes of payoff functions. For example, average performances can be evaluated by a mean-payoff function, peak performances by a limsup payoff function, and the parity payoff function can be used to encode logical specifications.
Surprisingly, all the MDPs equipped with mean, limsup or parity payoff functions share a common non-trivial property: they admit *pure stationary* optimal strategies.
In this paper, we introduce the class of *prefix-independent* and *submixing* payoff functions, and we prove that any MDP equipped with such a payoff function admits pure stationary optimal strategies.
This result unifies and simplifies several existing proofs. Moreover, it is a key tool for generating new examples of MDPs with pure stationary optimal strategies.

1 Introduction

Controller synthesis. One of the central questions in system theory is the controller synthesis problem : given a controllable system and a logical specification, is it possible to control the system so that its behaviour meets the specification?

In the most classical framework, the transitions of the system are not stochastic and the specification is given in LTL or CTL*. In that case, the controller synthesis problem reduces to computing a *winning strategy* in a parity game on graphs [Tho95].

There are two natural directions to extend this framework.

First direction consists in considering systems with stochastic transitions [dA97]. In that case the controller wishes to maximize the *probability* that the specification holds. The corresponding problem is the computation of an *optimal strategy* in a Markov decision process with parity condition [CY90].

* This research was supported by Instytut Informatyki of Warsaw University and European Research Training Network: Games and Automata for Synthesis and Validation.

Second direction to extend the classical framework of controller synthesis consists in considering quantitative specifications [dA98,CMH06]. Whereas a logical specification specifies good and bad behaviours of the system, a quantitative specification evaluates performances of the system in a more subtle way. These performances are evaluated by a *payoff function*, which associates a real value with each run of the system. Synthesis of a controller which maximizes performances of the system corresponds to the computation of an *optimal strategy* in a *payoff game* on graphs.

For example, consider a logical specification that specifies that the system should not reach an error state. Then using a payoff function, we can refine this logical specification. For example, we can specify that the number of visits to the error states is as small as possible, or also that the average time between two occurrences of the error state is as long as possible. Observe that logical specifications are a special case of quantitative specifications, where the payoff function takes only two possible values, 1 or 0, depending whether or not the behaviour of the system meets the specification.

In the most general case, the transitions of the system are stochastic and the specification is quantitative. In that case, the controller wishes to maximize the *expected value* of the payoff function, and the controller synthesis problem consists in computing an optimal strategy in a Markov decision process.

Positional payoff functions. Various payoff functions have been introduced and studied, in the framework of Markov decision processes but also in the broader framework of two player stochastic games. For example, the discounted payoff [Sha53,CMH06] and the total payoff [TV87] are used to evaluate short-term performances. Long-term performances can be computed using the mean-payoff [Gil57,dA98] or the limsup payoff [MS96] that evaluate respectively average performances and peak performances. These functions are central tools in economic modelization. In computer science, the most popular payoff function is the parity payoff function, which is used to encode logical properties.

Very surprisingly, the discounted, total, mean, limsup and parity payoff functions share a common non-trivial property. Indeed, in any Markov decision process equipped with one of those functions there exists optimal strategies of a very simple kind : they are at the same time *pure* and *stationary*. A strategy is pure when the controller plays in a deterministic way and it is stationary when choices of the controller depend only on the current state, and not on the full history of the run. For the sake of concision, pure stationary strategies are called *positional* strategies, and we say that a payoff function itself is positional if in any Markov decision process equipped with this function, there exists an optimal strategy which is positional.

The existence of positional optimal strategies has algorithmic interest. In fact, this property is the key for designing several polynomial time algorithms that compute values and optimal strategies in MDPs [Put94,FV97].

Recently, there has been growing research activity about the existence of positional optimal strategies in non-stochastic two-player games with infinitely many states [Grä04,CN06,Kop06] or finitely many states [BSV04,GZ05]. The

framework of this paper is different, since it deals with finite MDPs, i.e. one-player stochastic games with finitely many states and actions.

Our results. In this paper, we address the problem of finding a common property between the classical payoff functions introduced above, which explains why they are all positional. We give the following partial answer to that question.

First, we introduce the class of submixing payoff functions, and we prove that a payoff function which is submixing and prefix-independent is also positional (cf. Theorem 1).

This result partially solves our problem, since the parity, limsup and mean-payoff functions are prefix-independent and submixing (cf. Proposition 1).

Our result has several interesting consequences. First, it unifies and shortens disparate proofs of positionality for the parity [CY90], limsup [MS96] and mean [Bie87,NS03] payoff function (section 4). Second, it allows us to generate a bunch of new examples of positional payoff functions (section 5).

Plan. This paper is organized as follows. In section 2, we introduce notions of controllable Markov chain, payoff function, Markov decision process and optimal strategy. In section 3, we state our main result : prefix-independent and submixing payoff functions are positional (cf. Theorem 1). In the same section, we give elements of proof of Theorem 1. In section 4, we show that our main result unifies various disparate proofs of positionality. In section 5, we present new examples of positional payoff functions.

2 Markov decision processes

Let $\mathbf{S}$ be a finite set. The set of finite (resp. infinite) sequences on $\mathbf{S}$ is denoted $\mathbf{S}^*$ (resp. $\mathbf{S}^\omega$). A *probability distribution* on $\mathbf{S}$ is a function $\delta : \mathbf{S} \to \mathbb{R}$ such that $\forall s \in \mathbf{S},\ 0 \leq \delta(s) \leq 1$ and $\sum_{s\in\mathbf{S}} \delta(s) = 1$. The set of probability distributions on $\mathbf{S}$ is denoted $\mathcal{D}(\mathbf{S})$.

2.1 Controllable Markov chains and strategies

Definition 1. *A controllable Markov chain* $\mathcal{A} = (\mathbf{S}, \mathbf{A}, (\mathbf{A}(s))_{s\in\mathbf{S}}, p)$ *is composed of:*

- *a finite set of states* $\mathbf{S}$ *and a finite set of actions* $\mathbf{A}$*,*
- *for each state* $s \in \mathbf{S}$*, a set* $\mathbf{A}(s) \subseteq \mathbf{A}$ *of actions available in* s*,*
- *transition probabilities* $p : \mathbf{S} \times \mathbf{A} \to \mathcal{D}(\mathbf{S})$ *.*

When the current state of the chain is s, then the controller chooses an available action $a \in \mathbf{A}(s)$, and the new state is t with probability $p(t|s, a)$.

A triple $(s, a, t) \in \mathbf{S} \times \mathbf{A} \times \mathbf{S}$ such that $a \in \mathbf{A}(s)$ and $p(t|s, a) > 0$ is called a transition.

A *history* in $\mathcal{A}$ is an infinite sequence $h = s_0 a_1 s_1 \cdots \in \mathbf{S}(\mathbf{AS})^\omega$ such that for each n, (s_n, a_{n+1}, s_{n+1}) is a transition. State s_0 is called the source of h. The set of histories with source s is denoted $\mathbf{P}^\omega_{\mathcal{A},s}$. A *finite history* in $\mathcal{A}$ is a finite

sequence $h = s_0 a_1 \cdots a_{n-1} s_n \in \mathbf{S}(\mathbf{AS})^*$ such that for each n, (s_n, a_{n+1}, s_{n+1}) is a transition. s_0 is the source of h and s_n its target. The set of finite histories (resp. of finite histories with source s) is denoted $\mathbf{P}^*_{\mathcal{A}}$ (resp. $\mathbf{P}^*_{\mathcal{A},s}$).

A *strategy* in $\mathcal{A}$ is a function $\sigma : \mathbf{P}^*_{\mathcal{A}} \to \mathcal{D}(\mathbf{A})$ such that for any finite history $h \in \mathbf{P}^*_{\mathcal{A}}$ with target $t \in \mathbf{S}$, the distribution $\sigma(h)$ puts non-zero probabilities only on actions that are available in t, i.e. $(\sigma(h)(a) > 0) \implies (a \in \mathbf{A}(t))$. The set of strategies in $\mathcal{A}$ is denoted $\Sigma_{\mathcal{A}}$.

As explained in the introduction of this paper, certain types of strategies are of particular interest, such as *pure* and *stationary* strategies. A strategy is *pure* when the controller plays in a determnistic way, i.e. without using any dice, and it is *stationary* when the controller plays without using any memory, i.e. his choices only depend on the current state of the MDP, and not on the entire history of the play. Formally :

Definition 2. *A strategy $\sigma \in \Sigma_{\mathcal{A}}$ is said to be:*

- pure *if* $\forall h \in \mathbf{P}^*_{\mathcal{A}}, \quad (\sigma(h)(a) > 0) \implies (\sigma(h)(a) = 1)$,
- stationary *if* $\forall h \in \mathbf{P}^*_{\mathcal{A}}$ *with target* t, $\sigma(h) = \sigma(t)$,
- positional *if it is pure and stationary.*

Since the definition of a stationary strategy may be confusing, let us remark that $t \in \mathbf{S}$ denotes at the same time the target state of the finite history $h \in \mathbf{P}^*_{\mathcal{A}}$ and also the finite history $t \in \mathbf{P}^*_{\mathcal{A},t}$ of length 1.

2.2 Probability distribution induced by a strategy

Suppose that the controller uses some strategy σ and that transitions between states occur according to the transition probabilities specified by $p(\cdot|\cdot,\cdot)$. Then intuitively the finite history $s_0 a_1 \cdots a_n s_n$ occurs with probability

$$\sigma(s_0)(a_1) \cdot p(s_1|s_0, a_1) \cdots \sigma(s_0 \cdots s_{n-1})(a_n) \cdot p(s_n|s_{n-1}, a_n) \ .$$

In fact, it is also possible to measure probabilities of infinite histories. For this purpose, we equip $\mathbf{P}^{\omega}_{\mathcal{A},s}$ with a σ-field and a probability measure. For any finite history $h \in \mathbf{P}^*_{\mathcal{A},s}$, and action a, we define the sets of infinite plays with prefix h or ha:

$$\mathcal{O}_h = \{s_0 a_1 s_1 \cdots \in \mathbf{P}^{\omega}_{\mathcal{A},s} \mid \exists n \in \mathbb{N}, s_0 a_1 \cdots s_n = h\}$$
$$\mathcal{O}_{ha} = \{s_0 a_1 s_1 \cdots \in \mathbf{P}^{\omega}_{\mathcal{A},s} \mid \exists n \in \mathbb{N}, s_0 a_1 \cdots s_n a_{n+1} = ha\} \ .$$

$\mathbf{P}^{\omega}_{\mathcal{A},s}$ is equipped with the σ-field generated by the collection of sets $\mathcal{O}_h$ and $\mathcal{O}_{ha}$. In the sequel, a measurable set of infinite paths will be called an *event*. Moreover, when there is no risk of confusion, the events $\mathcal{O}_h$ and $\mathcal{O}_{ha}$ will be denoted simply h and ha.

A theorem of Ionescu Tulcea (cf. [BS78]) implies that there exists a unique probability measure $\mathbb{P}^{\sigma}_s$ on $\mathbf{P}^{\omega}_{\mathcal{A},s}$ such that for any finite history $h \in \mathbf{P}^*_{\mathcal{A},s}$ with

target t, and for every $a \in \mathbf{A}(t)$,

$$\mathbb{P}_s^\sigma(ha \mid h) = \sigma(h)(a) \ , \tag{1}$$
$$\mathbb{P}_s^\sigma(har \mid ha) = p(r|t,a) \ . \tag{2}$$

We will use the following random variables. For $n \in \mathbb{N}$, and $t \in \mathbf{S}$,

$$\begin{array}{ll} S_n(s_0a_1s_1\cdots) = s_n & \text{the } (n+1)\text{-th state,} \\ A_n(s_0a_1s_1\cdots) = a_n & \text{the } n\text{-th action,} \\ H_n = S_0A_1\cdots A_nS_n & \text{the finite history of the first } n \text{ stages,} \\ N_t = |\{n > 0 : S_n = t\}| \in \mathbb{N} \cup \{+\infty\} & \text{the number of visits to state } t. \end{array} \tag{3}$$

2.3 Payoff functions

After an infinite history of the controllable Markov chain, the controller gets some payoff. There are various ways for computing this payoff.

Mean payoff. The mean-payoff function has been introduced by Gilette [Gil57] and is used to evaluate average performance. Each transition (s,a,t) of the controllable Markov chain is labeled with a daily payoff $r(s,a,t) \in \mathbb{R}$. An history $s_0a_1s_1\cdots$ gives rise to a sequence $r_0r_1\cdots$ of daily payoffs, where $r_n = r(s_n, a_{n+1}, s_{n+1})$. The controller receives the following payoff:

$$\phi_{\text{mean}}(r_0r_1\cdots) = \limsup_{n\in\mathbb{N}} \frac{1}{n+1}\sum_{i=0}^{n} r_i \ . \tag{4}$$

Discounted payoff. The discounted payoff has been introduced by Shapley [Sha53] and is used to evaluate short-term performance. Each transition (s,a,t) is labeled not only with a daily payoff $r(s,a,t) \in \mathbb{R}$ but also with a discount factor $0 \leq \lambda(s,a,t) < 1$. The payoff associated with a sequence $(r_0,\lambda_0)(r_1,\lambda_1)\cdots \in (\mathbb{R}\times[0,1[)^\omega$ of daily payoffs and discount factors is:

$$\phi^\lambda_{\text{disc}}((r_0,\lambda_0)(r_1,\lambda_1)\cdots) = r_0 + \lambda_0 r_1 + \lambda_0\lambda_1 r_2 + \cdots \ . \tag{5}$$

Parity payoff. The parity payoff function is used to encode temporal logic properties [GTW02]. Each transition (s,a,t) is labeled with some priority $c(s,a,t) \in \{0,\ldots,d\}$. The controller receives payoff 1 if the highest priority seen infinitely often is odd, and 0 otherwise. For $c_0c_1\cdots \in \{0,\ldots,d\}^\omega$,

$$\phi_{\text{par}}(c_0c_1\cdots) = \begin{cases} 0 \text{ if } \limsup_n c_n \text{ is even,} \\ 1 \text{ otherwise.} \end{cases} \tag{6}$$

General payoffs. In the sequel, we will give other examples of payoff functions. Observe that in the examples we gave above, the transitions were labeled with various kinds of data: real numbers for the mean-payoff, couple of real numbers for the discounted payoff and integers for the parity payoff.

We wish to treat those examples in a unified framework. For this reason, we consider now that each controllable Markov chain $\mathcal{A}$ comes together with a finite set of colours $\mathbf{C}$ and a mapping $\mathrm{col} : \mathbf{S} \times \mathbf{A} \times \mathbf{S} \to \mathbf{C}$, which colors transitions.

In the case of the mean payoff, transitions are coloured with real numbers hence $\mathbf{C} \subseteq \mathbb{R}$, whereas in the case of the discounted payoff colours are couples $\mathbf{C} \subseteq \mathbb{R} \times [0, 1[$ and for the parity game colours are integers $\mathbf{C} = \{0, \dots, d\}$.

For an history (resp. a finite history) $h = s_0 a_1 s_1 \cdots$, the colour of the history h is the infinite (resp. finite) sequence of colours

$$\mathrm{col}(h) = \mathrm{col}(s_0, a_1, s_1)\, \mathrm{col}(s_1, a_2, s_2) \cdots \ .$$

Definition 3. *Let $\mathbf{C}$ be a finite set. A payoff function on $\mathbf{C}$ is a measurable*[1] *and bounded function $\phi : \mathbf{C}^\omega \to \mathbb{R}$.*

After an history h, the controller receives payoff $\phi(\mathrm{col}(h))$.

2.4 Values and optimal strategies in Markov decision processes

Definition 4. *A Markov decision process is a couple $(\mathcal{A}, \phi)$, where $\mathcal{A}$ is a controllable Markov chain coloured by a set $\mathbf{C}$ and ϕ is a payoff function on $\mathbf{C}$.*

Let us fix a Markov decision process $\mathcal{M} = (\mathcal{A}, \phi)$. After history h, the controller receives payoff $\phi(\mathrm{col}(h)) \in \mathbb{R}$. We extend the definition domain of ϕ to $\mathbf{P}^\omega_{\mathcal{A},s}$:

$$\forall h \in \mathbf{P}^\omega_{\mathcal{A},s}, \quad \phi(h) = \phi(\mathrm{col}(h)) \ .$$

The expected value of ϕ under the probability $\mathbb{P}^\sigma_s$ is called the *expected payoff* of the controller and is denoted $\mathbb{E}^\sigma_s[\phi]$. It is well-defined because ϕ is measurable and bounded. The *value of a state s* is the maximal expected payoff that the controller can get :

$$\mathrm{val}(\mathcal{M})(s) = \sup_{\sigma \in \Sigma_\mathcal{A}} \mathbb{E}^\sigma_s[\phi] \ .$$

A strategy σ is said to be *optimal* in $\mathcal{M}$ if for any state $s \in \mathbf{S}$,

$$\mathbb{E}^\sigma_s[\phi] = \mathrm{val}(\mathcal{M})(s) \ .$$

3 Optimal positional control

We are interested in those payoff functions that ensure the existence of positional optimal strategies. It motivates the following definition.

[1] relatively to the Borelian σ-field on $\mathbf{C}^\omega$.

Definition 5. *Let* $\mathbf{C}$ *be a finite set of colors and* ϕ *a payoff function on* $\mathbf{C}^\omega$*. Then* ϕ *is said to be* positional *if for any controllable Markov chain* $\mathcal{A}$ *coloured by* $\mathbf{C}$*, there exists a positional optimal strategy in the MDP* $(\mathcal{A}, \phi)$*.*

Our main result concerns the class of payoff functions with the following properties.

Definition 6. *Let* ϕ *be a payoff function on* $\mathbf{C}^\omega$*. We say that* ϕ *is* prefix-independent *if for any finite word* $u \in \mathbf{C}^*$ *and infinite word* $v \in \mathbf{C}^\omega$*,* $\phi(uv) = \phi(v)$*. See [Cha06] for interesting results about concurrent stochastic games with prefix-independent payoff functions. We say that* ϕ *is* submixing *if for any sequence of finite non-empty words* $u_0, v_0, u_1, v_1, \ldots \in \mathbf{C}^*$*,*

$$\phi(u_0 v_0 u_1 v_1 \cdots) \leq \max\{\ \phi(u_0 u_1 \cdots)\ ,\ \phi(v_0 v_1 \cdots)\ \}\quad .$$

The notion of prefix-independence is classical. The submixing property is close to the notions of *fairly-mixing* payoff functions introduced in [GZ04] and of *concave* winning conditions introduced in [Kop06]. We are now ready to state our main result.

Theorem 1. *Any prefix-independent and submixing payoff function is positional.*

The proof of this theorem is based on the 0-1 law and an induction on the number of actions. Due to space restrictions, we do not give details here, a full proof can be found in [Gim06].

4 Unification of classical results

We now show how Theorem 1 unifies proofs of positionality of the parity [CY90], the limsup and liminf [MS96] and the mean-payoff [Bie87,NS03] functions.

The parity, mean, limsup and liminf payoff functions are denoted respectively $\phi_{\text{par}}, \phi_{\text{mean}}, \phi_{\text{lsup}}$ and ϕ_{linf}. Both ϕ_{par} and ϕ_{mean} have already been defined in subsection 2.3. ϕ_{lsup} and ϕ_{linf} are defined as follows. Let $\mathbf{C} \subseteq \mathbb{R}$ be a finite set of real numbers, and $c_0 c_1 \cdots \in \mathbf{C}^\omega$. Then

$$\phi_{\text{lsup}}(c_0 c_1 \cdots) = \limsup_n c_n$$
$$\phi_{\text{linf}}(c_0 c_1 \cdots) = \liminf_n c_n\quad .$$

The four payoff functions $\phi_{\text{par}}, \phi_{\text{mean}}, \phi_{\text{lsup}}$ and ϕ_{linf} are very different. Indeed, ϕ_{lsup} measures the peak performances of the system, ϕ_{linf} the worst performances, and ϕ_{mean} the average performances. The function ϕ_{par} is used to encode logical specifications, expressed in MSO or LTL for example [GTW02].

Proposition 1. *The payoff functions* ϕ_{lsup}*,* ϕ_{linf}*,* ϕ_{par} *and* ϕ_{mean} *are submixing.*

Proof. Let $\mathbf{C} \subseteq \mathbb{R}$ be a finite set of real numbers and $u_0, v_0, u_1, v_1, \ldots \in \mathbf{C}^*$ be a sequence of finite non-empty words on $\mathbf{C}$. Define $u = u_0 u_1 \cdots \in \mathbf{C}^\omega$, $v = v_0 v_1 \cdots \in \mathbf{C}^\omega$ and $w = u_0 v_0 u_1 v_1 \cdots \in \mathbf{C}^\omega$. The following elementary fact immediately implies that ϕ_{lsup} is submixing.

$$\phi_{\text{lsup}}(w) = \max\{\phi_{\text{lsup}}(u), \phi_{\text{lsup}}(v)\} \ . \tag{7}$$

In a similar way, ϕ_{linf} is submixing since

$$\phi_{\text{linf}}(w) = \min\{\phi_{\text{linf}}(u), \phi_{\text{linf}}(v)\} \ . \tag{8}$$

Now suppose that $\mathbf{C} = \{0, \ldots, d\}$ is a finite set of integers and consider function ϕ_{par}. Remember that $\phi_{\text{par}}(w)$ equals 1 if $\phi_{\text{lsup}}(w)$ is odd and 0 if $\phi_{\text{lsup}}(w)$ is even. Then using (7) we get that if $\phi_{\text{par}}(w)$ has value 1 then it is the case of either $\phi_{\text{par}}(u)$ or $\phi_{\text{par}}(v)$. It proves that ϕ_{par} is also submixing.

Now let us consider function ϕ_{mean}. A proof that ϕ_{mean} is submixing already appeared in [GZ04], and we reproduce it here, updating the notations. Again $\mathbf{C} \subseteq \mathbb{R}$ is a finite set of real numbers. Let $c_0, c_1, \ldots \in \mathbf{C}$ be the sequence of letters of $\mathbf{C}$ such that $w = (c_i)_{i \in \mathbb{N}}$. Since word w is a shuffle of words u and v, there exists a partition (I_0, I_1) of $\mathbb{N}$ such that $u = (c_i)_{i \in I_0}$ and $v = (c_i)_{i \in I_1}$. For any $n \in \mathbb{N}$, let $I_0^n = I_0 \cap \{0, \ldots, n\}$ and $I_1^n = I_1 \cap \{0, \ldots, n\}$. Then for $n \in \mathbb{N}$,

$$\begin{aligned}\frac{1}{n+1}\sum_{i=0}^{n} c_i &= \frac{|I_0^n|}{n+1}\left(\frac{1}{|I_0^n|}\sum_{i \in I_0^n} c_i\right) + \frac{|I_1^n|}{n+1}\left(\frac{1}{|I_1^n|}\sum_{i \in I_1^n} c_i\right) \\ &\leq \max\left\{\frac{1}{|I_0^n|}\sum_{i \in I_0^n} c_i, \frac{1}{|I_1^n|}\sum_{i \in I_1^n} c_i\right\} \ .\end{aligned}$$

The inequality holds since $\frac{|I_0^n|}{n+1} + \frac{|I_1^n|}{n+1} = 1$. Taking the superior limit of this inequality, we obtain $\phi_{\text{mean}}(w) \leq \max\{\phi_{\text{mean}}(u), \phi_{\text{mean}}(v)\}$. It proves that ϕ_{mean} is submixing. □

Since ϕ_{lsup}, ϕ_{linf}, ϕ_{par} and ϕ_{mean} are clearly prefix-independent, Proposition 1 and Theorem 1 imply that those four payoff functions are positional. Hence, we unify and simplify existing proofs of [CY90,MS96] and [Bie87,NS03]. In particular, we use only elementary tools for proving the positionality of the mean-payoff function, whereas [Bie87] uses martingale theory and relies on other papers, and [NS03] uses a reduction to discounted games, as well as analytical tools.

5 Generating new examples of positional payoff functions.

We present three different techniques for generating new examples of positional payoff functions.

5.1 Mixing with the liminf payoff

In last section, we saw that peak performances of a system can be evaluated using the limsup payoff, whereas its worst performances are computed using the liminf payoff. The *compromise payoff* function is used when the controller wants to achieve a trade-off between good peak performances and not too bad worst performances. Following this idea, we introduced in [GZ04] the following payoff function. We fix a factor $\lambda \in [0,1]$, a finite set $\mathbf{C} \subseteq \mathbb{R}$ and for $u \in \mathbf{C}^\omega$, we define

$$\phi^\lambda_{\text{comp}}(u) = \lambda \cdot \phi_{\text{lsup}}(u) + (1-\lambda) \cdot \phi_{\text{linf}}(u) \ .$$

The fact that $\phi^\lambda_{\text{comp}}$ is submixing is a corollary of the following proposition.

Proposition 2. *Let $\mathbf{C} \subseteq \mathbb{R}$, $0 \leq \lambda \leq 1$ and ϕ be a payoff function on $\mathbf{C}$. Suppose that ϕ is prefix-independent and submixing. Then the payoff function*

$$\lambda \cdot \phi + (1-\lambda) \cdot \phi_{linf} \tag{9}$$

is also prefix-independent and submixing.

The proof is straightforward, using (8) above. According to Theorem 1 and Proposition 1, any payoff function defined by equation (9), where ϕ is either $\phi_{\text{mean}}, \phi_{\text{par}}$ or ϕ_{lsup}, is positional. Hence, this technique enable us to generate new examples of positional payoffs.

5.2 The approximation operator

Consider an increasing function $f : \mathbb{R} \to \mathbb{R}$ and a payoff function $\phi : \mathbf{C}^\omega \to \mathbb{R}$. Then their composition $f \circ \phi$ is also a payoff function and moreover, if ϕ is positional then $f \circ \phi$ also is. Indeed, a strategy optimal for an MDP $(\mathcal{A}, \phi)$ is also optimal for the MDP $(\mathcal{A}, f \circ \phi)$.

An example is the threshold function $f = \mathbf{1}_{\geq 0}$ which associates 0 with strictly negative real numbers and 1 with positive number. Then $f \circ \phi$ indicates whether the performance evaluated by ϕ reaches the critical value of 0.

Hence any increasing function $f : \mathbb{R} \to \mathbb{R}$ defines a unary operator on the family of payoff functions, and this operator stabilizes the family of positional payoff functions. In fact, it is straightforward to check that it also stabilizes the sub-family of prefix-independent and submixing payoff functions.

5.3 The hierarchical product

Now we define a binary operator between payoff functions, which also stabilizes the family of prefix-independent and submixing payoff functions. We call this operator the *hierarchical product*.

Let ϕ_0, ϕ_1 be two payoff functions on sets of colours $\mathbf{C}_0$ and $\mathbf{C}_1$ respectively. We do not require $\mathbf{C}_0$ and $\mathbf{C}_1$ to be identical nor disjoints.

The hierarchical product $\phi_0 \rhd \phi_1$ of ϕ_0 and ϕ_1 is a payoff function on the set of colours $\mathbf{C}_0 \cup \mathbf{C}_1$ and is defined as follows. Let $u = c_0 c_1 \cdots \in (\mathbf{C}_0 \cup \mathbf{C}_1)^\omega$ and u_0 and u_1 the two projections of u on $\mathbf{C}_0$ and $\mathbf{C}_1$ respectively. Then

$$(\phi_0 \rhd \phi_1)(u) = \begin{cases} \phi_0(u_0) & \text{if } u_0 \text{ is infinite,} \\ \phi_1(u_1) & \text{otherwise.} \end{cases}$$

This definition makes sense : although each word u_0 and u_1 can be either finite or infinite, at least one of them must be infinite.

Let us give examples of use of hierarchical product.

For $e \in \mathbb{N}$, let 0_e and 1_e be the payoff functions defined on the one-letter alphabet $\{e\}$ and constant equal to 0 and 1 respectively. Let d be an odd number, and ϕ_{par} be the parity payoff function on $\{0, \ldots, d\}$. Then

$$\phi_{\text{par}} = 1_d \rhd 0_{d-1} \rhd \cdots \rhd 1_1 \rhd 0_0 \ .$$

Another example of hierarchical product was given in [GZ05,GZ06], where we defined and establish properties about the *priority mean-payoff function*. This payoff function is in fact the hierarchical product of d mean-payoff functions. Remark that another way of fusionning the parity payoff and the mean-payoff functions has been presented in [CHJ05], and the resulting payoff function is not positional. In contrary, it turns out that the priority mean-payoff function is positional, as a corollary of Theorem 1, and the following proposition, whose proof is easy.

Proposition 3. *Let ϕ_0 and ϕ_1 be two payoff functions. If ϕ_0 and ϕ_1 are prefix-independent and submixing, then $\phi_0 \rhd \phi_1$ also is.*

5.4 Towards a quantitative specification language?

In the previous section, we defined two unary operators and one binary operator over payoff functions. Moreover, we proved that the class of prefix-independent and submixing payoff functions is stable under these operators. As a consequence, if we start with the constant, the limsup, the liminf and the mean payoff functions, and we apply recursively our three operators, we get a huge family of sub-mixinf and prefix-independent payoff functions. According to Theorem 1, all those functions are positional.

We hope that this result is a first step towards a rich quantitative specification language. For example, using the hierarchical product, we can express properties such as: "Minimize the frequency of visits to error states. In the case where error states are visited only finitely often, maximize the peak performances." The positionality of those payoff functions gives hope that the corresponding controller synthesis problems are solvable in polynomial time.

6 Conclusion

In that paper, we have introduced the class of prefix-independent and submixing payoff functions, and we proved that they are positional. Moreover, we have defined three operators on payoff functions, that can be used to generate new examples of MDPs with positional optimal strategies.

There are different natural directions to continue this work.

First, most of the results of this paper can be extended to the broader framework of two-player zero-sum stochastic games with full information. This is ongoing work with Wiesław Zielonka, to be published soon.

Second, the results of the last section give rise to natural algorithmic questions. For MDPs equipped with mean, limsup, liminf, parity or discounted payoff functions, the existence of optimal positional strategies is the key for designing algorithms that compute values and optimal strategies in polynomial time [FV97]. For examples generated with the mixing operator and the hierarchical product, it seems that values and optimal strategies are computable in exponential time, but we do not know the exact complexity. Also it is not clear how to obtain efficient algorithms when payoff functions are defined using approximation operators.

To conclude, let us formulate the following conjecture about positional payoff functions. "Any payoff function which is positional for the class of non-stochastic one-player games is positional for the class of Markov decision processes".

Acknoledgments

I would like to thank Wiesław Zielonka for numerous discussions about payoff games on MDP's.

References

[Bie87] K.-J. Bierth. An expected average reward criterion. *Stochastic Processes and Applications*, 26:133–140, 1987.

[BS78] D. Bertsekas and S. Shreve. *Stochastic Optimal Control: The Discrete-Time Case*. Academic Press, 1978.

[BSV04] H. Björklund, S. Sandberg, and S. Vorobyov. Memoryless determinacy of parity and mean payoff games: a simple proof, 2004.

[Cha06] K. Chatterjee. Concurrent games with tail objectives. In *CSL'06*, 2006.

[CHJ05] K. Chatterjee, T. A. Henzinger, and M. Jurdzinski. Mean-payoff parity games. In *LICS'05*, pages 178–187, 2005.

[CMH06] K. Chatterjee, R. Majumdar, and T. A. Henzinger. Markov decision processes with multiple objectives. In *STACS'06*, pages 325–336, 2006.

[CN06] T. Colcombet and D. Niwinski. On the positional determinacy of edge-labeled games. *Theor. Comput. Sci.*, 352(1-3):190–196, 2006.

[CY90] C. Courcoubetis and M. Yannakakis. Markov decision processes and regular events. In *ICALP'90*, volume 443 of *LNCS*, pages 336–349. Springer, 1990.

[dA97] L. de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, december 1997.

[dA98] L. de Alfaro. How to specify and verify the long-run average behavior of probabilistic systems. In *LICS*, pages 454–465, 1998.

[FV97] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer, 1997.

[Gil57] D. Gilette. Stochastic games with zero stop probabilities, 1957.

[Gim06] H. Gimbert. *Pure stationary optimal strategies in Markov decision processes.* Research Report 2006-02, Université Denis Diderot, LIAFA, 2006.

[Grä04] E. Grädel. Positional determinacy of infinite games. In *Proc. of STACS'04*, volume 2996 of *LNCS*, pages 4–18, 2004.

[GTW02] E. Grdel, W. Thomas, and T. Wilke. *Automata, Logics and Infinite Games*, volume 2500 of *LNCS*. Springer, 2002.

[GZ04] H. Gimbert and W. Zielonka. When can you play positionally? In *Proc. of MFCS'04*, volume 3153 of *LNCS*, pages 686–697. Springer, 2004.

[GZ05] H. Gimbert and W. Zielonka. Games where you can play optimally without any memory. In *CONCUR 2005*, volume 3653 of *LNCS*, pages 428–442. Springer, 2005.

[GZ06] H. Gimbert and W. Zielonka. Deterministic priority mean-payoff games as limits of discounted games. In *Proc. of ICALP 06*, LNCS. Springer, 2006.

[Kop06] E. Kopczyński. Half-positional determinacy of infinite games. In *Proc. of ICALP'06*, LNCS. Springer, 2006.

[MS96] A.P. Maitra and W.D. Sudderth. *Discrete gambling and stochastic games*. Springer-Verlag, 1996.

[NS03] A. Neyman and S. Sorin. *Stochastic games and applications*. Kluwer Academic Publishers, 2003.

[Put94] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1994.

[Sha53] L. S. Shapley. Stochastic games. In *Proceedings of the National Academy of Science USA*, volume 39, pages 1095–1100, 1953.

[Tho95] W. Thomas. On the synthesis of strategies in infinite games. In *Proc. of STACS'95,LNCS*, volume 900, pages 1–13, 1995.

[TV87] F. Thuijsman and O. J. Vrieze. *The Bad Match, a total reward stochastic game*, volume 9. 1987.