Expected Window Mean-Payoff

Benjamin Bordais
ENS Rennes, France

Shibashis Guha
Université libre de Bruxelles, Belgium

Jean-François Raskin
Université libre de Bruxelles, Belgium

Abstract

We study the expected value of the window mean-payoff measure in Markov decision processes (MDPs) and Markov chains (MCs). The window mean-payoff measure strengthens the classical mean-payoff measure by measuring the mean-payoff over a window of bounded length that slides along an infinite path. This measure ensures better stability properties than the classical mean-payoff. Window mean-payoff has been introduced previously for two-player zero-sum games. As in the case of games, we study several variants of this definition: the measure can be defined to be prefix-independent or not, and for a fixed window length or for a window length that is left parametric. For fixed window length, we provide polynomial time algorithms for the prefix-independent version for both MDPs and MCs. When the length is left parametric, the problem of computing the expected value on MDPs is as hard as computing the mean-payoff value in two-player zero-sum games, a problem for which it is not known if it can be solved in polynomial time. For the prefix-dependent version, surprisingly, the expected window mean-payoff value cannot be computed in polynomial time unless P=PSPACE. For the parametric case and the prefix-dependent case, we manage to obtain algorithms with better complexities for MCs.

2012 ACM Subject Classification Mathematics of computing → Stochastic processes; Mathematics of computing → Probability and statistics

Keywords and phrases mean-payoff, Markov decision processes, synthesis

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2019.32

Related Version A full version of the paper is available at `https://arxiv.org/abs/1812.09298`.

Funding Work partially supported by the ARC project *Non-Zero Sum Game Graphs: Applications to Reactive Synthesis and Beyond* (Fédération Wallonie-Bruxelles), and the EOS project *Verifying Learning Artificial Intelligence Systems* (F.R.S.-FNRS & FWO).

1 Introduction

Markov Decision processes (MDPs) are a classical model for decision-making in stochastic environments [16, 1]. Objectives in MDPs are formalized by functions that map infinite paths to values. Classical examples of such functions are the mean-payoff and the discounted sum [16]. The mean-payoff function does not guarantee local stability of the values along the path: if the mean-value of an infinite path is v, it is possible that for arbitrarily long infixes of the path, the mean-payoff of the infix is largely away from v. There have been several recent contributions [8, 4, 9, 5] that address this problem. Here, we study *window mean-payoff* objectives for MDPs; these objectives were first introduced in [8, 9] for two-player games.

In window mean-payoff [9], payoffs are considered over a local finite length window that slides along the path: the objective is to ensure that the mean-payoff always reaches a given threshold within the window length ℓ. This is a strengthening of classical mean-payoff: for all lengths ℓ, and all infinite sequences π of payoffs, if π satisfies the window mean-payoff objective for threshold v, then π has a mean-payoff of at least v. Interestingly, this additional stability property can always be met at the cost of a small degradation of mean-payoff performances in two-player games: whenever there exists a strategy with mean-payoff value

39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2019).
Editors: Arkadev Chattopadhyay and Paul Gastin; Article No. 32; pp. 32:1–32:15
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Table 1 Complexity, hardness and memory requirements for solving different window objectives for Markov decision processes and Markov chains (2-p stands for "two-player").

	MDP			Markov chain
	Complexity	**Memory**	**Hardness**	**Complexity**
WMP	polynomial (Thm. 5)	polynomial	2-p DirWMP (Thm. 9)	polynomial (Cor. of Thm. 5)
BWMP	UP ∩ coUP (Thm. 10)	memoryless	2-p Mean-payoff (Thm. 13)	polynomial[1] (Thm. 19)
DirWMP	exponential[2] (Thm. 15)	exponential	PSPACE (Thm. 16)	pseudopolynomial[3] (Thm. 21)

v then for every $\epsilon > 0$, there is a window length ℓ and a strategy that ensure that the window mean-payoff for threshold $v - \epsilon$ is eventually satisfied for windows of length ℓ (see Lemma 2(b) in [9]).

Here, we study how to maximize the expected value of the window mean-payoff function $\mathsf{f}^{\ell}_{\mathsf{DirWMP}}$ defined as follows: let $\pi : \mathbb{N} \to \mathbb{Z}$ be an infinite sequence of payoffs, then

$$\mathsf{f}^{\ell}_{\mathsf{DirWMP}}(\pi) = \sup\{\lambda \in \mathbb{R} \mid \forall i \in \mathbb{N} : \max_{1 \leq j \leq \ell} \frac{1}{j} \sum_{k=0}^{j-1} \pi(i+k) \geq \lambda\}$$

i.e., it returns the supremum of all thresholds that are enforced by the sequence of payoffs π for every window of length ℓ. As in [13], we study natural variants: (i) when the length of the window is fixed or it is left unspecified but needs to be bounded, and (ii) when the window property needs to be enforced from the beginning or not (leading to a prefix-independent variant.)

Main contributions. First, we provide an algorithm to compute the best expected value of $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ (prefix-independent version with fixed window length ℓ - noted WMP) with a time complexity polynomial in the size of the MDP and in ℓ (Theorem 5). As window mean-payoff objectives aim at strong stability over reasonable periods of time, it is natural to assume that ℓ is bounded polynomially by the size of the MDP, and so our algorithm is fully polynomial for those interesting cases. This complexity matches the complexity of computing the value of the function $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ for two-player games [9], and we provide a relative hardness result: deciding the existence of a winning strategy in a window mean-payoff game can be reduced in log-space to the problem of the expected value of $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ in an MDP (Theorem 9). Second, we consider the case in which the length ℓ in the measure $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ is not fixed but is required to be bounded (BWMP). We provide an algorithm in UP ∩ coUP (Theorem 10), and we show that providing a polynomial time solution for this case would give a polynomial time solution to the value problem in mean-payoff games (Theorem 13), a long-standing open problem [18]. Third, we consider the prefix-dependent version (DirWMP), i.e. the window property needs to hold directly from the beginning of the path. Surprisingly, this problem is expected to be harder: no polynomial time solution can exist unless P=PSPACE. Indeed, we show that this problem is PSPACE-HARD even if ℓ is given in unary (Theorem 16). We also provide an algorithm that executes in time that is polynomial in the size of the MDP, and in the largest weight appearing in the MDP, and exponential in the window length ℓ (Theorem 15). Finally, while our main results concentrate on MDPs, we also systematically provide results for the special case of Markov chains. An overview of our results is given in Table 1.

[1] independent of any window size
[2] exponential in window size and the number of bits to represent the weights on the edges
[3] pseudopolynomial in the number of bits to represent the weights on the edges

Related works. Window mean-payoff objectives were first introduced in [8] for two-player games, then for games with imperfect information in [13], and in combination with ω-regular constraints in [7]. Here, we consider them for MDPs instead of games. Still, we show that, for the prefix-independent version of the window objectives, inside an end-component of an MDP, the expected window mean-payoff value is closely related to the worst-case value of the associated zero-sum games (see Lemma 6 and Lemma 11). Stability issues of the mean-payoff measure triggered other works. First, in [4], MDPs with the objective to optimize the expected mean-payoff performance and stability are studied. Their notion of stability is related to statistical variance. The notion of stability offered by window mean-payoff objective studied here, is stronger. The techniques used to solve the two problems differ: in [4] they rely on solving sets of quadratic constraints, while our techniques rely on graph game algorithms and linear programming. Second, [5] introduces *window-stability objectives*. They are directly inspired from the window mean-payoff objective of [4] but contrary to window mean-payoff objectives, do not enjoy the inductive window property which is heavily used in our algorithms. Also, [5] considers games (2 players) and graphs (1 player) but not MDPs ($1\frac{1}{2}$ players).

MDP with classical mean-payoff objectives have been studied both for the threshold probability problem, in which the objective is to find a strategy that maximizes the probability that the mean-payoff is above a given threshold, and for the expectation problem that asks for a strategy that maximizes the expected value of the mean-payoff [16]. Combination of both types of constraints have been considered in [3]. The work of Brihaye et al. [6], appeared recently on arXiv, and was done independently of our work. The authors of [6] consider the threshold probability problem for window mean-payoff objectives in MDP: given a threshold $\lambda \in \mathbb{Q}$, and a window length ℓ, the problem asks to find a strategy that maximizes the probability of obtaining a window mean-payoff greater than or equal to λ. We study the expectation problem, and as for traditional mean-payoff objectives, the two problems are different and cannot be easily reduced to one another (see [3] and the discussion in the previous paragraph). Our work and their work are largely complementary. Some of the basic techniques employed in the two papers are however similar, e.g. for the prefix-independent objectives, both the works analyze maximal end components in related ways. Nevertheless, there are also important differences between the two works; e.g. we show that for the expected value, the prefix-dependent and the prefix-independent versions of the bounded window mean-payoff objective, lead to the same value; this is not the case for the threshold probability problem. We also show interesting connections between the fixed and the bounded case, for the expected value problem: the bounded case can be seen as the limit of the fixed case (Theorem 14), again this property does not hold for the threshold problem. Also, the algorithms for direct fixed window objective differ largely for the two problems. Though both the problems have been shown to be PSpace-Hard, the expected value problem requires a more involved reduction. Finally, while we have shown how to solve the expected value problem for the special case of MCs for which we establish better complexity results, this is not considered in [6].

Structure of the paper. Sect. 2 introduces the necessary definitions and concepts. Sect. 3 defines the different variants of window mean-payoff objectives. Sect. 4 studies the prefix-independent variants while Sect. 5 covers the prefix-dependent variants. Algorithms and hardness results are given for all the problems. Finally, Sect. 6 considers the special case of MCs. We only provide sketches of the proofs here. Full proofs are given in [2].

2 Preliminaries

For $k \in \mathbb{N}$, we denote by $[k]_0$ and $[k]$ the set of natural numbers $\{0, \ldots, k\}$ and $\{1, \ldots, k\}$ respectively. Given a finite set A, a (rational) *probability distribution* over A is a function $\mathsf{Pr}\colon A \to [0,1] \cap \mathbb{Q}$ such that $\sum_{a \in A} \mathsf{Pr}(a) = 1$. We denote the set of probability distributions on A by $\mathcal{D}(A)$. The *support* of a probability distribution Pr on A is $\mathsf{Supp}(\mathsf{Pr}) = \{a \in A \mid \mathsf{Pr}(a) > 0\}$, and Pr is called *Dirac* if $|\mathsf{Supp}(\mathsf{Pr})| = 1$. An event is said to happen *almost surely* if it happens with probability 1.

Markov chain. A weighted *Markov chain* (MC, for short) is a tuple $\mathcal{M} = \langle S, E, s_{\mathsf{init}}, w, \mathbb{P} \rangle$, where S is a set of states, $s_{\mathsf{init}} \in S$ is an initial state, $E \subseteq S \times S$ is a set of edges, $w : E \to \mathbb{Q}$ maps edges to *weights* (or *payoff*), and $\mathbb{P} : S \to \mathcal{D}(E)$ assigns a probability distribution on the set $E(s)$ of outgoing edges from s. In the following, $\mathbb{P}(s, (s, s'))$ is denoted $\mathbb{P}(s, s')$, for all $s \in S$. The Markov chain $\mathcal{M}$ is *finite* if S is finite.

For $s \in S$, the set of *infinite paths* in $\mathcal{M}$ starting from s is $\mathsf{Paths}^{\mathcal{M}}(s) = \{\pi = s_0 s_1 \ldots \in S^\omega \mid s_0 = s, \forall n \in \mathbb{N}, \mathbb{P}(s_n, s_{n+1}) > 0\}$. The set of all the paths in $\mathcal{M}$ is $\mathsf{Paths}^{\mathcal{M}} = \bigcup_{s \in S} \mathsf{Paths}^{\mathcal{M}}(s)$. For a path $\pi = s_0 s_1 \ldots \in \mathsf{Paths}^{\mathcal{M}}$, by $\pi(i, l)$ we denote the sequence of $l+1$ states (or l edges) $s_i \ldots s_{i+l}$, and for simplicity, we denote $\pi(i, 0)$ by $\pi(i)$. The infinite suffix of π starting in s_n is denoted by $\pi(n, \infty) \in \mathsf{Paths}^{\mathcal{M}}$. The set of *finite* paths starting from a state $s \in S$ is defined as $\mathsf{Fpaths}^{\mathcal{M}}(s) = \{\pi = s \ldots s' \in S^+ \mid \exists \overline{\pi} \in \mathsf{Paths}^{\mathcal{M}},\ \pi\overline{\pi} \in \mathsf{Paths}^{\mathcal{M}}(s)\}$ and $\mathsf{Fpaths}^{\mathcal{M}} = \bigcup_{s \in S} \mathsf{Fpaths}^{\mathcal{M}}(s)$. For $\pi = s \ldots s'$, we denote by $\mathsf{Last}(\pi)$, the last state s' in π.

Consider some measurable function $\mathsf{f} : \mathsf{Paths}^{\mathcal{M}}(s_{\mathsf{init}}) \to \mathbb{R}$ associating a value to each infinite path starting from s_{init}. For an interval $I \subseteq \mathbb{R}$, we denote by $\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, I)$ the set $\{\pi \in \mathsf{Paths}^{\mathcal{M}}(s_{\mathsf{init}}) \mid \mathsf{f}(\pi) \in I\}$, and for $r \in \mathbb{R}$, we denote by $\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, r)$ the set $\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, [r, r])$. Since the set of paths $\mathsf{Paths}^{\mathcal{M}}(s_{\mathsf{init}})$ forms a probability space, measured by a function Pr [17], and f is a random variable, we denote by $\mathbb{E}^{\mathcal{M}}_{s_{\mathsf{init}}}(\mathsf{f}) = \int_{x \in \mathbb{R}} \mathsf{Pr}(\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, x)) \cdot x$ the *expected value* of f over the set of paths starting from s_{init}.

The *bottom strongly connected components* (BSCCs for short) in a finite Markov chain $\mathcal{M}$ are the strongly connected components $\mathcal{B}$ from which it is impossible to exit, i.e. for all $s \in \mathcal{B}$ and $t \in \mathcal{M}$, if $\mathbb{P}(s, t) > 0$ then $t \in \mathcal{B}$. We denote by $\mathsf{BSCC}(\mathcal{M})$ the set of BSCCs of $\mathcal{M}$. Every infinite path eventually ends up in one of the BSCCs with probability 1. Considering $\Diamond$ and $\Box$ as the standard LTL *eventually* and *always* operators and that $\Diamond\Box\mathcal{B}$ denotes that eventually the path visits only states in $\mathcal{B}$ (see [1] for a formal definition), we formally state:

▶ **Proposition 1.** *For all $s \in S$, $\mathsf{Pr}(\pi \in \mathsf{Paths}^{\mathcal{M}}(s) \mid \exists \mathcal{B} \in \mathsf{BSCC}(\mathcal{M}),\ \pi \models \Diamond\Box\mathcal{B}) = 1$.*

Markov decision process. A finite weighted *Markov decision process* (MDP, for short) is a tuple $\Gamma = \langle S, E, \mathit{Act}, s_{\mathsf{init}}, w, \mathbb{P} \rangle$, where S is a finite set of states, $s_{\mathsf{init}} \in S$ is an initial state, Act is a finite set of actions, and $E \subseteq S \times \mathit{Act} \times S$ is a set of edges, the function $w : E \to \mathbb{Q}$ maps edges to *weights* (or *payoffs*), and $\mathbb{P} : S \times \mathit{Act} \to \mathcal{D}(E)$ is a function that assigns a probability distribution on the set $E(s, a)$ of outgoing edges from s if action $a \in \mathit{Act}$ is taken from s. Given $s \in S$ and $a \in \mathit{Act}$, we define $\mathsf{Post}(s, a) = \{s' \in S \mid \mathbb{P}(s, a)(s, s') > 0\}$. Then, for all states $s \in S$, we denote by $\mathit{Act}(s)$ the set of actions $\{a \in \mathit{Act} \mid \mathsf{Post}(s, a) \neq \emptyset\}$. We assume that, for all $s \in S$, we have $\mathit{Act}(s) \neq \emptyset$. In the following, we denote $\mathbb{P}(s, a)(s, s')$ by $\mathbb{P}(s, a, s')$.

A *strategy* in Γ is a function $\sigma : S^+ \to \mathcal{D}(\mathit{Act})$ such that $\mathsf{Supp}(\sigma(s_0 \ldots s_n)) \subseteq \mathit{Act}(s_n)$, for all $s_0 \ldots s_n \in S^+$. We denote by $\mathsf{strat}(\Gamma)$ the set of strategies available in Γ. Once we fix a strategy σ in an MDP $\Gamma = \langle S, E, \mathit{Act}, s_{\mathsf{init}}, w, \mathbb{P} \rangle$, we obtain an MC $\Gamma^{[\sigma]}$ [1]. A strategy σ is *deterministic*, if for each $s_0 \ldots s_n \in S^+$, the distribution assigned by σ is Dirac, otherwise the strategy is *randomized*. We show that deterministic strategies suffice for playing optimally in all the problems considered here. For a sequence $\rho \in S^+$ of states, we also denote by

$\mathsf{Last}(\rho)$ the last state in ρ. Consider a measurable function f that associates a value to infinite paths in Markov chains. Then, we call $\sup_{\sigma \in \mathsf{strat}(\Gamma)} \mathbb{E}^{\Gamma^{[\sigma]}}_{s_{\mathsf{init}}}(\mathsf{f})$ the optimal expected value of f in Γ. In the sequel, when clear from the context, we denote $\sup_{\sigma \in \mathsf{strat}(\Gamma)} \mathbb{E}^{\Gamma^{[\sigma]}}_{s_{\mathsf{init}}}(\mathsf{f})$ by $\mathbb{E}^{\Gamma}_{s_{\mathsf{init}}}(\mathsf{f})$. A deterministic strategy can be encoded by a transition system $\langle Q, act, \delta, \iota \rangle$ where Q is a (possibly infinite) set of states, commonly called modes, $act : Q \times S \to Act$ selects an action such that, for all $q \in Q$ and $s \in S$, $act(q, s) \in Act(s)$, $\delta : Q \times S \to Q$ is a mode update function and $\iota : S \to Q$ selects an initial mode for each state $s \in S$. The amount of memory used by such a strategy is defined to be $|Q|$. A strategy is said to be *memoryless* if $|Q| = 1$, that is, the choice of action only depends on the current state where the choice is made. Formally, a strategy is memoryless if for all finite sequences of states ρ_1 and ρ_2 in S^+ such that $\mathsf{Last}(\rho_1) = \mathsf{Last}(\rho_2)$, we have $\sigma(\rho_1) = \sigma(\rho_2)$. A strategy is called *finite memory* if Q is finite. Note that the state space of $\Gamma^{[\sigma]}$ is $S \times Q$. For a sequence π of states in $\Gamma^{[\sigma]}$, we denote by $\mathsf{proj}(\pi)_{|S}$ the corresponding sequence of states in the MDP Γ.

An *end-component* (EC, for short) $M = (T, A)$ with $T \subseteq S$, and $A : T \to 2^{Act}$ is a *sub-MDP* of Γ (for all $s \in T$, we have $A(s) \subseteq Act(s)$, and for all $a \in A(s)$, we have $\mathsf{Post}(s, a) \subseteq T$) that is strongly connected. A *maximal EC* (MEC, for short) is an EC that is not included in any other EC. We denote by $\mathsf{MEC}(\Gamma)$ the set of all maximal end components of Γ. Any infinite path will eventually end up in one maximal end component almost surely, whatever strategy is considered. This is stated in the following proposition:

▶ **Proposition 2** ([11]). *In an MDP Γ, for each strategy $\sigma \in \mathsf{strat}(\Gamma)$, for every state $s \in S$, and mode $q \in Q$, we have:* $\mathsf{Pr}(\pi \in \mathsf{Paths}^{\Gamma^{[\sigma]}}(s, q) \mid \exists M = (T, A) \in \mathsf{MEC}(\Gamma), \mathsf{proj}(\pi)_{|S} \models \Diamond\Box T) = 1$.

Weighted two-player games. An MDP can also be considered to have the semantics of a two-player turn-based game (denoted 2P) played for infinitely many rounds while ignoring the probabilities. Every 2P we consider here can be played optimally with deterministic strategies, therefore we restrict ourselves to deterministic strategies for both players. The first round starts from s_{init}. In each round, Player 1 chooses an action $a \in Act(s)$ from a state s while Player 2 chooses a state $s' \in \mathsf{Post}(s, a)$. We denote by $G_\Gamma = \langle S, E, Act, s_{\mathsf{init}}, w \rangle$ the two-player game that is obtained from an MDP $\Gamma = \langle S, E, Act, s_{\mathsf{init}}, w, \mathbb{P} \rangle$.

For a 2P, Player 1 thus chooses among the deterministic strategies available in MDPs. A strategy of Player 2 is a function $\mu : S^+ \cdot Act \to S$, with the restriction that if $\mu(s_0 s_1 \ldots s_n \cdot a) = s$ then $\mathbb{P}(s_n, a, s) > 0$. The set of deterministic strategies for Player 1 and Player 2 are denoted by $\mathsf{strat}_1(G)$ and $\mathsf{strat}_2(G)$ respectively. In a two-player game there is no randomness: Given two strategies $\sigma_1 \in \mathsf{strat}_1(G)$ and $\sigma_2 \in \mathsf{strat}_2(G)$, we denote by $\pi_{(G,s,\sigma_1,\sigma_2)}$ the unique path that occurs in 2P G under strategies σ_1 and σ_2 from state s. Then, for a function f that associates a value to each infinite path, we denote by $V^{\mathsf{f}}_s(G)$ the value $\sup\limits_{\sigma_1 \in \mathsf{strat}_1(G)} \inf\limits_{\sigma_2 \in \mathsf{strat}_2(G)} \mathsf{f}(\pi_{(G,s,\sigma_1,\sigma_2)})$. The definitions of the memories of strategies also apply to two-player games.

In the following, in MCs, MDPs and in 2Ps, w.l.o.g. we consider only non-negative integer weights[4]. We denote by W the maximum weight appearing on the edges for MCs, MDPs and 2Ps. We denote the size of an MC $\mathcal{M}$, MDP Γ and 2P G by $|\mathcal{M}|$, $|\Gamma|$ and $|G|$ respectively. This size is equal to $|S| + |E|$.

[4] For weights belonging to $\mathbb{Q}$, we can multiply them with the LCM d of their denominators to obtain integer weights. Among the resultant set of integer weights, if the minimum integer weight κ is negative, then we add -κ to the weight of each edge so that the resultant weights are natural numbers. For a function f if the expected value was originally x, then the new expected value is $d \cdot x - \kappa$.

3 Window Mean-Payoff Value

Let $\mathcal{M} = \langle S, E, s_{\mathsf{init}}, w, \mathbb{P}\rangle$ be a finite MC. Let $\rho = s_0 \dots s_n \in \mathsf{Fpaths}^{\mathcal{M}}(s)$, we define $\mathsf{MP} : \mathsf{Fpaths}^{\mathcal{M}} \to \mathbb{Q}$ as: $\mathsf{MP}(\rho) = \frac{1}{n}\sum_{i=0}^{n} w(s_i, s_{i+1})$, where $n = |\rho| > 0$, the number of edges in ρ. For $\pi = s_0 \dots \in \mathsf{Paths}^{\mathcal{M}}$, the *mean-payoff* function $\mathsf{f}_{\mathsf{Mean}} : \mathsf{Paths}^{\mathcal{M}} \to \mathbb{R}$ is defined as

$$\mathsf{f}_{\mathsf{Mean}}(\pi) = \liminf_{n\to\infty} \mathsf{MP}(s_0 \dots s_n) \tag{1}$$

We now define several variants of *window mean-payoff value functions*. For $\pi = s_0 s_1 \dots s_n \dots \in \mathsf{Paths}^{\mathcal{M}}$, a window size ℓ, and a position i, the window mean-payoff value of π in position i over length ℓ is defined by $\mathsf{WMP}^{\ell}(\pi(i,\infty)) = \max_{k\in[\ell]} \mathsf{MP}(\pi(i,k))$, i.e. it is the maximal value of the mean-payoff of an infix of π that starts at position i and with a size at most ℓ. For a threshold λ such that $\mathsf{WMP}^{\ell}(\pi(i,\infty)) \geq \lambda$, we say that the window mean-payoff value over length ℓ is at least λ at position i. We define the *fixed window mean-payoff function* $\mathsf{f}^{\ell}_{\mathsf{WMP}} : \mathsf{Paths}^{\mathcal{M}} \to \mathbb{R}$ such that, for every path $\pi = s_0 s_1 \dots s_n \dots \in \mathsf{Paths}^{\mathcal{M}}$:

$$\mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi) = \sup\{\lambda \in \mathbb{R} \mid \exists k \in \mathbb{N},\ \forall i \geq k : \mathsf{WMP}^{\ell}(\pi(i,\infty)) \geq \lambda\} \tag{2}$$

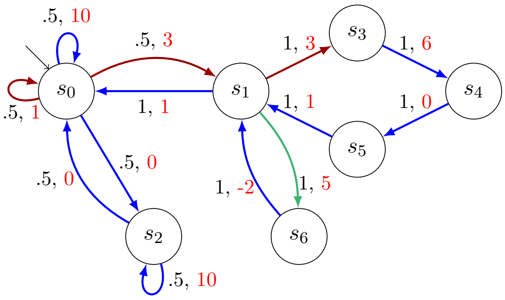

Figure 1 In the MEC M with initial state s_0, the expected value of $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ (resp. $\mathsf{f}_{\mathsf{BWMP}}$) is the maximum of the value of the two-player game with the direct fixed window mean-payoff (resp. classical mean-payoff) objective obtained over all states. Also, for $\ell = 3$, we have $\mathbb{E}^{M}_{s_0}(\mathsf{f}^{\ell}_{\mathsf{WMP}}) < \mathbb{E}^{M}_{s_0}(\mathsf{f}_{\mathsf{BWMP}}) < \mathbb{E}^{M}_{s_0}(\mathsf{f}_{\mathsf{Mean}})$.

The value $\mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi)$ corresponds to the supremum over all thresholds λ where for every such λ, there exists a position k such that for all positions $i \geq k$, the window mean-payoff value over length ℓ is at least λ. We note some properties of the function $\mathsf{f}^{\ell}_{\mathsf{WMP}}$. First, it is prefix-independent, that is, for every path $\pi \in \mathsf{Paths}^{\mathcal{M}}$, for all $n \geq 1$, we have $\mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi) = \mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi(n,\infty))$. Second, it is a strengthening of the classical mean-payoff function: for all paths π, we have that $\mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi) \leq \mathsf{f}_{\mathsf{Mean}}(\pi)$. And finally, $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ imposes strong stability properties: if $\mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi) \geq \lambda$, then from some point on in π, it is always the case that the observed mean-payoff from position i gets larger than λ within position $i + \ell$. This stability property is not enforced by classical mean-payoff function for which infixes of arbitrary lengths can have arbitrary low mean-payoffs.

Then, we define the *bounded window mean-payoff function* $\mathsf{f}_{\mathsf{BWMP}} : \mathsf{Paths}^{\mathcal{M}} \to \mathbb{R}$ such that, for every path $\pi = s_0 \dots \in \mathsf{Paths}^{\mathcal{M}}$:

$$\mathsf{f}_{\mathsf{BWMP}}(\pi) = \sup\{\lambda \in \mathbb{R} \mid \exists \ell, k \geq 1, \forall i \geq k : \mathsf{WMP}^{\ell}(\pi(i,\infty)) \geq \lambda\} \tag{3}$$

Here, the length of the window is not fixed but it needs to be bounded.

Now, we define the *direct fixed window mean-payoff function* $\mathsf{f}^{\ell}_{\mathsf{DirWMP}} : \mathsf{Paths}^{\mathcal{M}} \to \mathbb{R}$ such that, for every path $\pi = s_0 \ldots \in \mathsf{Paths}^{\mathcal{M}}$:

$$\mathsf{f}^{\ell}_{\mathsf{DirWMP}}(\pi) = \sup\{\lambda \in \mathbb{R} \mid \forall i \geq 0 : \mathsf{WMP}^{\ell}(\pi(i,\infty)) \geq \lambda\} \tag{4}$$

Here the window property must hold from the beginning of the path and so it is not prefix-independent. For every path $\pi \in \mathsf{Paths}^{\mathcal{M}}$, we have $\mathsf{f}^{\ell}_{\mathsf{DirWMP}}(\pi) \leq \mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi)$. Finally, we define the *direct bounded window mean-payoff function* $\mathsf{f}_{\mathsf{DirBWMP}} : \mathsf{Paths}^{\mathcal{M}} \to \mathbb{R}$ such that, for every path $\pi = s_0 \ldots \in \mathsf{Paths}^{\mathcal{M}}$:

$$\mathsf{f}_{\mathsf{DirBWMP}}(\pi) = \sup\{\lambda \in \mathbb{R} \mid \exists \ell \geq 1, \forall i \geq 0 : \mathsf{WMP}^{\ell}(\pi(i,\infty)) \geq \lambda\} \tag{5}$$

i.e., variant where the length of the window is not fixed.

The following proposition relates some of the variants defined above in a Markov chain $\mathcal{M}$.

▶ **Proposition 3.** *Let $\pi \in \mathsf{Paths}^{\mathcal{M}}$. Then, we have: $\sup_{\ell \geq 1} \mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi) = \mathsf{f}_{\mathsf{BWMP}}(\pi) \leq \mathsf{f}_{\mathsf{Mean}}(\pi)$.*

▶ **Example 4.** Consider the example in Figure 1 where the MDP Γ is a single MEC. The probabilities appear in black and the weights in red. The strategy that chooses the blue action in s_0 and in s_2 maximizes the expected value of the classical mean-payoff function $\mathsf{f}_{\mathsf{Mean}}$ in Γ from s_0. The expected value of this strategy is 5. However, clearly, while playing this strategy, we run the risk of having a mean-payoff of 0 for arbitrarily long period (while looping between s_0 and s_2). So it may not be the best strategy if we aim at some stability property in the mean-payoff. In this example, the strategy that maximizes the expected value of $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ for $\ell = 3$, is the strategy that plays the brown action in state s_0 and then alternates between the brown and green action in s_1.

4 Algorithms and Hardness for Prefix-independent Objectives

The fixed window mean-payoff function for length ℓ can be solved in time that is polynomial in the size of the MDP and in ℓ:

▶ **Theorem 5.** *Given an MDP Γ with maximum weight W, a window length ℓ and a threshold $\lambda \in \mathbb{Q}$, whether $\mathbb{E}^{\Gamma}_{s_{init}}(\mathsf{f}^{\ell}_{\mathsf{WMP}}) \geq \lambda$ can be decided in $O(\mathsf{poly}(|\Gamma|, \ell, \log_2 W))$ time and deterministic polynomial memory strategies suffice to play optimally.*

To establish this result, we first study the case of a single MEC $M = (T, A)$. By Proposition 2, for every strategy σ, each path of $\Gamma^{[\sigma]}$ will almost surely end up in an MEC. Since $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ is prefix-independent, the value of a path only depends on its behavior in the MEC in which it ends up. Since M is strongly connected (as it is an MEC), for every $s, s' \in T$, there exists a strategy $\sigma_{(s,s')} \in \mathsf{strat}(M)$ such that every path starting from s reaches s' almost surely, in the Markov chain $M^{[\sigma_{(s,s')}]}$. Therefore, for all $s, s' \in T$, we have $\mathbb{E}^{\Gamma}_{s}(\mathsf{f}^{\ell}_{\mathsf{WMP}}) = \mathbb{E}^{\Gamma}_{s'}(\mathsf{f}^{\ell}_{\mathsf{WMP}})$, i.e. the optimal expected value is the same from all states in the MEC. We denote by λ^{ℓ}_{M} this optimal value. Now, the following lemma interestingly relates λ^{ℓ}_{M} to the maximum over all states s of the optimal *adversarial value* from s (which is the value of $V_s^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}$), that is when the stochastic behavior in M is replaced by an adversary:

▶ **Lemma 6.** *Let $M = (T, A)$ be an MEC that is also an MDP. Then $\lambda^{\ell}_{M} = \max\limits_{s \in T} V_s^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}(G_M)$.*

Proof sketch. Let $v \in T$ be a state that maximizes the value of the 2P that is, $V_v^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}(G_M) = \max_{s \in T} V_s^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}(G_M)$. When a strategy σ is fixed in M, using classical probability arguments (Borel-Cantelli) every possible finite sequence of states (with respect to the strategy σ) is visited infinitely often almost surely. In particular, the worst sequence of states in terms of maximizing the fixed window mean-payoff (that is the sequence that Player 2 chooses in the two-player game G_M) is visited infinitely often almost surely. Hence, the expected window mean-payoff in the MEC M is at most the value of the 2P G_M from v, that is $V_v^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}(G_M)$.

Now, consider a strategy in the MEC M that consists in reaching v and then playing according to an optimal deterministic strategy of Player 1 in the two-player game G_M from v. Then, every path in M consistent with that strategy has a window mean-payoff of at least $V_v^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}(G_M)$. Thus the expected value of the window mean-payoff is at least $V_v^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}(G_M)$. ◀

To solve the two-player game, we rely on the following result from [9]:

▶ **Theorem 7.** *Given a* 2P *with maximum weight W, a window length ℓ, and a threshold $\lambda \in \mathbb{Q}$, in a two-player window mean-payoff game, for both the fixed window and the direct fixed window mean-payoff objectives, it can be decided in* $O(\mathsf{poly}(|G|, \ell, \log_2 W))$ *time if Player* 1 *has a winning strategy. For both players, an optimal strategy may need memory that is linear in $|G|$ and ℓ and such strategies can be constructed in time* $O(\mathsf{poly}(|G|, \ell, \log_2 W))$*, and deterministic strategies suffice to play optimally.*

▶ **Example 8.** Consider again the example of Figure 1. Lemma 6 tells us that we need to compute the two-player game value of the direct fixed window objective for $\ell = 3$ at each state of the MEC. We can check that this value is equal to 2 for all states but s_0 and s_2 in which the game values are equal to 1 and 2/3 respectively. Now, to obtain the best expected value for $\mathsf{f}^{\ell}_{\mathsf{WMP}}$ with $\ell = 3$ from s_0, we must play a strategy that first reaches almost surely any state $s \notin \{s_0, s_2\}$ and then switches to an optimal strategy for the two-player game from s.

As we know how to deal with an MEC, we now consider the general case.

Proof sketch of Theorem 5. Our algorithm for solving the general case proceeds as follows: (i) it decomposes Γ into MECs, (ii) for each MEC M, it computes the value λ_M^{ℓ} as described in Lemma 6, (iii) it constructs a new MDP Γ^{MEC} that is identical to Γ except that every MEC $M \in \mathsf{MEC}(\Gamma)$ is now compacted into a single state s_M, the transition relation is defined accordingly to mimic the transition relation of Γ over its MECs, the value of each transition that self-loops on s_M is assigned the value λ_M^{ℓ}, as computed in point (ii), and the other transitions have the same value as in Γ, (iv) it computes the optimal expected (classical) mean-payoff value for the new MDP Γ^{MEC}. It should be clear that the optimal expected mean-payoff of Γ^{MEC} is equal to the optimal expected window mean-payoff value in Γ.

Now we analyze the complexity of this algorithm. The MEC decomposition of Γ of step (i) can be done in quadratic time [10] in the size of Γ yielding at most $|S|$ MECs. By Theorem 7, given a threshold λ, for every MEC M and for each state s in M, it can be decided in time $O(\mathsf{poly}(|M| \cdot \ell \cdot \log_2 W))$ whether Player 1 has a winning strategy for the direct fixed window mean-payoff game from s. To find the maximal expected window mean-payoff in M, we do a binary search over a set $\Lambda = \{\frac{p}{q} \mid q \in [\ell],\ p \in [q \cdot W]_0\}$ with $|\Lambda| = O(W \cdot \ell^2)$ different possible values of λ and decide the two-player game starting from each state in M for each such λ. Furthermore, the construction of Γ^{MEC} can be done in time $O(|\Gamma|)$. Finally, the maximal expected value of the (classical) mean-payoff in Γ^{MEC} can be computed in polynomial time (see e.g. [16]) using linear programming. Thus all the steps can be done in time $O(\mathsf{poly}(|\Gamma|, \ell, \log_2 W))$.

We construct the optimal strategy σ from steps $(i) - (iii)$ by combining them with a deterministic memoryless strategy that optimizes the expected value of the (classical) mean-payoff in Γ^{MEC} (step (iv)). When this memoryless strategy prescribes to stay in an MEC M, we apply inside M the strategy defined in the proof of Lemma 6. The memory used by the strategy σ is polynomial in $|\Gamma|$ and ℓ as announced. ◀

The algorithm above relies on solving two-player games for the direct fixed window mean-payoff objective. We next show that this step cannot be improved without improving the algorithms for those games. Indeed, the following relative hardness result holds: solving the two-player game for the direct fixed window mean-payoff objective can be reduced in log-space to computing the expected value of the fixed window mean-payoff function.

▶ **Theorem 9.** *Given a* 2P *G with an initial state s_{init} and a window length ℓ, we can construct in log-space an MDP Γ_G with an initial state s'_{init} such that $\mathbb{E}^{\Gamma_G}_{s'_{\mathit{init}}}(\mathsf{f}^{\ell}_{\mathsf{WMP}}) = V^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}_{s_{\mathit{init}}}(G)$.*

Proof sketch. Consider a weighted two-player game $G = \langle S, E, \mathit{Act}, s_{\mathsf{init}}, w\rangle$. We construct an MDP Γ from G such that $\mathbb{E}^{\Gamma}_{s_{\mathsf{init}}}(\mathsf{f}^{\ell}_{\mathsf{WMP}}) = V^{\mathsf{f}^{\ell}_{\mathsf{DirWMP}}}_{s_{\mathsf{init}}}(G)$.

We first construct another game $G^{reset} = \langle S, E', \mathit{Act}, s_{\mathsf{init}}, w'\rangle$ from G where $E' = E \cup \{(s, a, s_{\mathsf{init}}) \mid (s, a, s_{\mathsf{init}}) \notin E, s \in S \setminus \{s_{\mathsf{init}}\}$ and $a \in \mathit{Act}(s)\}$ and $w'(e) = w(e)$ for $e \in E$ and $w'(e) = (W + 1) \cdot \ell$ for $e \in E' \setminus E$. Note that the game graph of G^{reset} is strongly connected. In the game G^{reset}, since Player 2 may "reset" the game at any time by taking an edge to s_{init}, the maximum of the value over all starting states of the two-player game is achieved at the state s_{init}. Moreover, the weight on these new edges being high enough, it is not in the interest of Player 2 to take one of them more than once. It follows that the values of the two-player game, starting from s_{init}, for the direct fixed window objective are the same in G and G^{reset}.

Now considering G^{reset} as an MDP $\Gamma = \langle S, E', \mathit{Act}, s_{\mathsf{init}}, w', \mathbb{P}\rangle$, such that for all $e \in E'$, we have $\mathbb{P}(e) > 0$, we note that Γ is actually an MEC. The result follows from Lemma 6. ◀

We now consider the prefix-independent version of the bounded window mean-payoff objective. For that case, we provide a UP $\cap$ coUP solution.

▶ **Theorem 10.** *Given an MDP Γ and a threshold $\lambda \in \mathbb{Q}$, deciding whether $\mathbb{E}^{\Gamma}_{s_{\mathit{init}}}(\mathsf{f}_{\mathsf{BWMP}}) \geq \lambda$ is in* UP $\cap$ coUP *and deterministic memoryless strategies suffice to play optimally.*

Since $\mathsf{f}_{\mathsf{BWMP}}$ is prefix-independent, similar to the fixed case, we first consider a single MEC M. All the states in the MEC M have the same value λ_M, and surprisingly, this value is the maximum over all states s of M of the optimal adversarial value from s (i.e. when the stochastic behavior is replaced by an adversary), for the *classical mean-payoff value* $V^{\mathsf{f}_{\mathsf{Mean}}}_s(G_M)$:

▶ **Lemma 11.** *Let $M = (T, A)$ be an MEC that is also an MDP. Then $\lambda_M = \max\limits_{s \in T} V^{\mathsf{f}_{\mathsf{Mean}}}_s(G_M)$.*

Proof sketch. Let v be a state such that $V^{\mathsf{f}_{\mathsf{Mean}}}_v(G_M) = \max_{s\in T} V^{\mathsf{f}_{\mathsf{Mean}}}_s(G_M)$.

Consider an optimal strategy σ_2 for Player 2 (that can be chosen among deterministic memoryless strategies). Now, let $\sigma \in \mathsf{strat}(M)$ (note that σ may be a randomised strategy), $\ell \geq 1$ and s be a state in the Markov chain $M^{[\sigma]}$. Any path π compatible with strategies σ and σ_2 must ensure $V^{\mathsf{f}_{\mathsf{Mean}}}_v(G_M) \geq \mathsf{f}_{\mathsf{Mean}}(\pi) \geq \mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi)$ (the last inequality is given by Proposition 3). Hence, in the MC $M^{[\sigma]}$, there is a non-zero probability to reach a sequence of states whose window mean-payoff is below $V^{\mathsf{f}_{\mathsf{Mean}}}_v(G_M)$ from s. This is true for every state s in $M^{[\sigma]}$. It follows that for every path $\pi \in \mathsf{Paths}^{M^{[\sigma]}}$, almost surely a sequence of states whose window mean-payoff is at most $V_v^{\mathsf{f}_{\mathsf{Mean}}}(G_M)$ is visited infinitely often. Therefore, the fixed window mean-payoff for length ℓ of a path in $M^{[\sigma]}$ is almost surely at most $V_v^{\mathsf{f}_{\mathsf{Mean}}}(G_M)$. This is true for every $\ell \geq 1$. Hence, by Proposition 3, we have that the bounded window mean-payoff of a path in $M^{[\sigma]}$ is at most $V_v^{\mathsf{f}_{\mathsf{Mean}}}(G_M)$ almost surely. Thus, $\mathbb{E}^{M[\sigma]}(\mathsf{f}_{\mathsf{BWMP}}) \leq V_v^{\mathsf{f}_{\mathsf{Mean}}}(G_M)$. This holds for every strategy $\sigma \in \mathsf{strat}(M)$. Therefore, $\lambda_M \leq V_v^{\mathsf{f}_{\mathsf{Mean}}}(G_M)$.

Now, consider an optimal strategy $\sigma_1 \in \mathsf{strat}_1(G_M)$ for Player 1 in G_M. Let ρ be a cycle of mean-payoff m that is minimal among all the cycles compatible with σ_1. Note that $V_v^{\mathsf{f}_{\mathsf{Mean}}}(G_M)$ equals m. Every path $\pi \in \mathsf{Paths}^{M^{[\sigma_1]}}$ has a bounded window mean-payoff of at least m since every cycle appearing in π has a mean-payoff of at least m, and for every ε, there exists $\ell > 0$ such that a direct fixed window mean-payoff of $m - \varepsilon$ can be ensured for every window of length ℓ along π. Thus $\lambda_M \geq m = \max_{s \in T} V_s^{\mathsf{f}_{\mathsf{Mean}}}(G_M)$ and hence the result. ◀

▶ **Example 12.** Consider again the example of Figure 1. Lemma 11 tells us that we need to compute the two-player game value of the classical mean-payoff objective $\mathsf{f}_{\mathsf{Mean}}$ at each state of the MEC. We can check that this value is equal to 2.5 for all states (by taking the brown action from s_1) but s_0 and s_2 at which the game value is equal to 1 . Now, to obtain the best expected value for $\mathsf{f}_{\mathsf{BWMP}}$, we must play a strategy that first reaches almost surely, from s_0, any other state $s \notin \{s_0, s_2\}$, (e.g. always play brown) and then switches to the optimal strategy for the two-player game from s for the classical mean-payoff objective.

We can now prove our main theorem for the bounded window mean-payoff objective.

Proof sketch for Theorem 10. The algorithm for this case follows exactly the algorithm in four steps (i), (ii), (iii), and (iv) of the algorithm for the proof of Theorem 5, with the difference, that step (ii) computes λ_M instead of λ_M^ℓ, and we use Lemma 11 to this end. The complexity of the algorithm is no more polynomial but in UP ∩ coUP because step (ii) requires solving a mean-payoff game [18, 14]. To construct an optimal strategy, we follow the same recipe as in the proof of Theorem 5. In this case, the strategies are deterministic and memoryless (mean-payoff games can be played optimally with memoryless strategies) and so deterministic memoryless strategies are sufficient to obtain the optimal expected value of the function $\mathsf{f}_{\mathsf{BWMP}}$. ◀

The following theorem shows that a polynomial time solution to our problem would lead to a polynomial time algorithm to solve mean-payoff games. The proof uses a reduction similar to the one used in the proof of Theorem 9.

▶ **Theorem 13.** *Given a two-player game G with an initial state s_{init}, we can construct in log-space an MDP Γ_G with an initial state s'_{init} such that $\mathbb{E}^{\Gamma_G}_{s'_{\mathit{init}}}(\mathsf{f}_{\mathsf{BWMP}}) = V_{s_{\mathit{init}}}^{\mathsf{f}_{\mathsf{Mean}}}(G)$.*

Finally, we show that in an MDP, the expected bounded window mean-payoff equals the supremum of the fixed window mean-payoff over all window lengths and over all strategies, which match the intuition behind these definitions.

▶ **Theorem 14.** *For every MDP Γ, we have* $\sup_{\sigma \in \mathsf{strat}(\Gamma)} \mathbb{E}^{\Gamma[\sigma]}(\mathsf{f}_{\mathsf{BWMP}}) = \sup_{\ell} \sup_{\sigma \in \mathsf{strat}(\Gamma)} \mathbb{E}^{\Gamma[\sigma]}(\mathsf{f}^{\ell}_{\mathsf{WMP}})$.

5 Algorithms and Hardness for Direct Variants

We start with the direct fixed window objective. Surprisingly the complexity of solving this objective is substantially higher than its prefix-independent conterpart. Our algorithm is exponential in ℓ and in the number of bits to encode W. As shown later, the higher complexity is explained by the fact that the problem is PSpace-Hard.

▶ **Theorem 15.** *Given an MDP Γ with an initial state s_{init}, a window length ℓ and a threshold $\lambda \in \mathbb{Q}$, whether $\mathbb{E}^{\Gamma}_{s_{init}}(\mathsf{f}^{\ell}_{\mathsf{DirWMP}}) \geq \lambda$ can be decided in time $O(\mathsf{poly}(|S| \cdot W^{\ell} \cdot \ell^2))$ and deterministic exponential memory strategies suffice to play optimally.*

Proof sketch. As $\mathsf{f}^{\ell}_{\mathsf{DirWMP}}$ is prefix-dependent, it is not sufficient to know the expected value of this function in the MECs of Γ. Instead, we construct a new MDP Γ_ℓ which is a finite state structure that maps each infinite path π of Γ to the minimal mean-payoff encountered in a window of size ℓ along this path. The state space of Γ_ℓ is $S' = S \times ([W]_0)^{\ell-1} \times \Lambda$ where $\Lambda = \{\frac{p}{q} \mid q \in [\ell],\ p \in [q \cdot W]_0\}$ and the initial state $s'_{\mathsf{init}} = (s_{\mathsf{init}}, [W, \dots, W], W)$. Informally, a state $t = (s, [w_1, \dots, w_{\ell-1}], \lambda_t) \in S'$ summarizes all finite paths $\rho = s_0 \dots s$ in Γ where the last $\ell - 1$ weights encountered are $w_1, \dots, w_{\ell-1}$, and λ_t keeps track of the minimum window mean-payoff seen so far in π for window size ℓ. Moreover, in MDP Γ_ℓ every edge exiting t has a weight equal to λ_t. In this way, for each $\pi' \in \mathsf{Paths}^{\Gamma_\ell}$, the sequence of weights seen along π' is a non-increasing series of values belonging to the finite set Λ. Thus, eventually the sequence reaches a value λ which never changes again, this λ is the direct fixed window mean-payoff of the corresponding path in Γ and because every edge exiting t has a weight equal to λ_t, we see that λ is also the mean-payoff of π' in Γ_ℓ. Now, it remains to compute the optimal expected mean-payoff in Γ_ℓ which can be done in polynomial time in the size of Γ_ℓ using linear programming, see e.g.[16]. This optimal expected mean-payoff in Γ_ℓ is equal to the optimal expected direct fixed window mean-payoff for window size ℓ in Γ. Note that although the algorithm is exponential in ℓ and in the number of bits used to represent W, it is fixed parameter tractable, if we consider W and ℓ as parameters.

Since optimal expected mean-payoff in an MDP can be achieved using memoryless deterministic strategies and the size of Γ_ℓ is exponential in the size of the original MDP Γ, an optimal strategy with memory exponential in the size of Γ exists. ◀

We now provide the following hardness result:

▶ **Theorem 16.** *Given an MDP Γ with an initial state s_{init}, a window length ℓ and a $\lambda \in \mathbb{Q}$, deciding whether $\mathbb{E}^{\Gamma}_{s_{init}}(\mathsf{f}^{\ell}_{\mathsf{DirWMP}}) \geq \lambda$ is* PSpace-Hard*.*

Proof. We show a reduction from the threshold probability problem for shortest path objectives [12]. An instance of the threshold probability problem is given by an MDP $\Gamma = (S, E, Act, s_{\mathsf{init}}, w, \mathbb{P})$ where w.l.o.g., we have that w assigns positive weights on the edges, $T \subseteq S$ is a set of target states, and for a strategy σ, the truncated sum $TS^T : Paths(\Gamma^{[\sigma]}) \longrightarrow \mathbb{N} \cup \infty$ up to T from the initial state s_{init} is defined as

$$TS^T(\rho) = \begin{cases} \sum_{i=0}^{n-1} w(e_i) & \text{if } \exists n \text{ such that } \rho(n) \in T \text{ and } \forall i \leq n-1, \text{ we have } \rho(i) \notin T \\ \infty & \text{if } \forall i \geq 0, \rho(i) \notin T, \end{cases}$$

where $e_i = (\rho(i), a, \rho(i+1))$, $a \in Act$; for a threshold $L \in \mathbb{N}$, and a probability threshold p, the problem asks to decide if there exists a strategy σ such that $\mathbb{P}_{\Gamma^{[\sigma]}, s_{\mathsf{init}}}[\{\rho \in Paths(\Gamma^{[\sigma]}) \mid TS^T(\rho) \leq L\}] \geq p$. The problem is known to be PSpace-Complete, even for acyclic MDPs [12]. The target set T is assumed to be made of absorbing states (i.e., with self-loops); the acyclicity is to be interpreted over the rest of the underlying graph.

Let $\Gamma = (S, E, Act, s_{\mathsf{init}}, w, \mathbb{P})$, where $S = T \uplus V$, and T is a set of target vertices. The acyclicity of that MDP implies that, from the initial state $s_{\mathsf{init}} \notin T$, it takes at most $|S| - 1$ steps to reach a vertex in T. Let W be the maximum weight appearing in Γ. We assume that $L \leq W \cdot (|S| - 1)$, otherwise the problem is trivial.

We construct a new MDP $\Gamma' = (S', E, Act', s_{\mathsf{init}}, w', \mathbb{P}')$ where $S' = S \cup \{s_{\mathsf{final}_1}, s_{\mathsf{final}_2}\}$, $Act' = Act \cup \{\mathsf{loop}, \alpha, \beta\}$. The set of edges $E' = \{(v, a, s) \mid (v, a, s) \in E,\ v \in V,\ s \in S\} \cup \{(t, \alpha, s_{\mathsf{final}_1}) \mid t \in T\} \cup \{(t, \beta, s_{\mathsf{final}_2}) \mid t \in T\} \cup \{(s_{\mathsf{final}_1}, \mathsf{loop}, s_{\mathsf{final}_1})\} \cup \{(s_{\mathsf{final}_2}, \mathsf{loop}, s_{\mathsf{final}_2})\}$ $\cup \{(v, \beta, s_{\mathsf{final}_2}) \mid v \in V$ and there is no outgoing edge from v in $\Gamma\}$. The probability function $\mathbb{P}'$ is defined as:

- $\mathbb{P}'(v, a, s) = \mathbb{P}(v, a, s)$ such that $(v, a, s) \in E,\ v \in V,\ s \in S$;
- $\mathbb{P}'(t, \alpha, s_{\mathsf{final}_1}) = 1$ for $t \in T$;
- $\mathbb{P}'(t, \beta, s_{\mathsf{final}_2}) = 1$ for $t \in T$;
- $\mathbb{P}'(s_{\mathsf{final}_j}, \mathsf{loop}, s_{\mathsf{final}_j}) = 1$ for $j \in \{1, 2\}$;
- $\mathbb{P}'(v, \beta, s_{\mathsf{final}_2}) = 1$ for $(v, \beta, s_{\mathsf{final}_2}) \in E'$ and $v \in V$;

The weight function w' is defined as follows.

- $w'(v, a, s) = -w(v, a, s)$ such that $(v, a, s) \in E,\ v \in V,\ s \in S$;
- $w'(t, \alpha, s_{\mathsf{final}_1}) = L$, for $t \in T$;
- $w'(t, \beta, s_{\mathsf{final}_2}) = W \cdot (|S| - 1)$, for $t \in T$;
- $w'(s_{\mathsf{final}_1}, \mathsf{loop}, s_{\mathsf{final}_1}) = 0$;
- $w'(s_{\mathsf{final}_2}, \mathsf{loop}, s_{\mathsf{final}_2}) = -\frac{1}{|S|}$, and
- $w'(v, \beta, s_{\mathsf{final}_2}) = W \cdot (|S| - 1)$ for $(v, \beta, s_{\mathsf{final}_2}) \in E'$ and $v \in V$.

Let $\ell = |S|$. Starting from s_{init}, since the weights on all the edges on the paths leading to a state in $t \in T$ are negative, the direct fixed window mean-payoff will consider paths until they reach s_{final_j} for $j \in \{1, 2\}$ given that the weights on the edges outgoing from t are positive.

We now call a path to be *good* if t appears in the path for some $t \in T$, and the sum of the edges from s_{init} to t is at least $-L$, otherwise the path is *bad*. Note that for a good path, choosing α leads to a direct fixed window mean-payoff of 0, while choosing β leads to direct fixed window mean-payoff of $-\frac{1}{|S|}$. On the other hand, for a bad path, choosing α gives a direct fixed window mean-payoff of at most $-\frac{1}{|S|}$, while choosing β gives a direct fixed window mean-payoff of $-\frac{1}{|S|}$. Therefore, for an optimal strategy, the direct fixed window mean-payoff for a *good* path is 0, and for a *bad* path, it is $-\frac{1}{|S|}$.

We have $|\Gamma'| = O(\mathsf{poly}(|\Gamma|))$. Furthermore, the expected value of the direct fixed window mean-payoff, $\mathbb{E}^{\Gamma}_{s_{\mathsf{init}}}(\mathsf{f}^{\ell}_{\mathsf{DirWMP}}) \geq p \cdot 0 + (1 - p) \cdot -\frac{1}{|S|} = -(1 - p) \cdot \frac{1}{|S|}$ iff there is a solution to the threshold probability problem.

Note that since $\ell = |S|$, deciding whether the expected value of the direct fixed window mean-payoff for an MDP is greater than or equal to some threshold is PSPACE-HARD even when ℓ is given in unary. Thus, we cannot expect to have an algorithm that is polynomial in the value of ℓ unless P=PSPACE[5]. ◀

We now consider the bounded case. In fact, the function $\mathsf{f}_{\mathsf{DirBWMP}}$ is equivalent to $\mathsf{f}_{\mathsf{BWMP}}$:

▶ **Lemma 17.** *For every path π in an MDP, we have that* $\mathsf{f}_{\mathsf{DirBWMP}}(\pi) = \mathsf{f}_{\mathsf{BWMP}}(\pi)$.

Proof sketch. It is easy to see that $\mathsf{f}_{\mathsf{DirBWMP}}(\pi) \leq \mathsf{f}_{\mathsf{BWMP}}(\pi)$. Now for every $\varepsilon > 0$, a window mean-payoff value of $\mathsf{f}_{\mathsf{BWMP}}(\pi) - \varepsilon$ can be ensured from the beginning of the path π by considering appropriately large window length. Since $\mathsf{f}_{\mathsf{DirBWMP}}(\pi)$ is the supremum of the window mean-payoff values that can be ensured with arbitrarily large window lengths, the result follows. ◀

[5] The reduction does not work for Markov chains since we cannot get a threshold for the window mean-payoff that separates the cases when there is a solution to the threshold probability problem for shortest path objective and when a solution to the problem does not exist. That is, if the sum of path from s'_{init} to t is below L and the edge corresponding to action α is taken in t, we do not know how much below 0 will the window mean-payoff be.

As a direct corollary of Lemma 17, Theorem 10 and Theorem 13, we obtain:

▶ **Theorem 18.** *Given an MDP Γ and a $\lambda \in \mathbb{Q}$, we have $\mathbb{E}^{\Gamma}_{s_{init}}(\mathsf{f}_{\mathsf{DirBWMP}}) = \mathbb{E}^{\Gamma}_{s_{init}}(\mathsf{f}_{\mathsf{BWMP}})$, and whether $\mathbb{E}^{\Gamma}_{s_{init}}(\mathsf{f}_{\mathsf{DirBWMP}}) \geq \lambda$ can be decided in* UP $\cap$ coUP*, and it is as hard as solving two-player mean-payoff games.*

6 Solving Window Mean-Payoff Objectives for Markov Chain

We focus on the bounded window objective and the direct fixed window objective for MCs, as MCs are special cases of MDPs, and for these two objectives, we show strict improvement in the complexity of the algorithms compared to MDPs. We start with the bounded window mean-payoff function, for which we provide a polynomial time solution while the case of MDPs is at least as hard as mean-payoff games (Theorem 13).

▶ **Theorem 19.** *Given an MC $\mathcal{M}$ and a threshold $\lambda \in \mathbb{Q}$, whether $\mathbb{E}^{\mathcal{M}}_{s_{init}}(\mathsf{f}_{\mathsf{BWMP}}) \geq \lambda$ can be decided in polynomial time.*

We first outline the case of a BSCC $\mathcal{B}$ since by Proposition 1, each path in an MC almost surely ends up in a BSCC. Let $\lambda_{\mathcal{B}}$ be the expected value of $\mathsf{f}_{\mathsf{BWMP}}$ in $\mathcal{B}$:

▶ **Lemma 20.** *For an MC that is a BSCC $\mathcal{B}$, we have $\lambda_{\mathcal{B}} = \min\limits_{\rho \in \mathsf{ElemCycles}(\mathcal{B})} \mathsf{MP}(\rho)$.*

Proof sketch. For every ℓ, a path π in $\mathcal{B}$ will almost surely have infinitely many infixes of length ℓ going around a minimum mean-cycle, leading to $\mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi) \leq c_{\mathcal{B}}$ where $c_{\mathcal{B}} = \min\limits_{\rho \in \mathsf{ElemCycles}(\mathcal{B})} \mathsf{MP}(\rho)$. Moreover, for each path π in $\mathcal{B}$ and for every $\varepsilon > 0$, by choosing an appropriate window length ℓ, we have $\mathsf{f}^{\ell}_{\mathsf{WMP}}(\pi) \geq c_{\mathcal{B}} - \varepsilon$. By definition of $\mathsf{f}_{\mathsf{BWMP}}$, we have $\mathsf{f}_{\mathsf{BWMP}}(\pi) = c_{\mathcal{B}}$ almost surely. ◀

Proof sketch of Theorem 19. Note that $\mathbb{E}^{\mathcal{M}}_{s_{\mathsf{init}}}(\mathsf{f}_{\mathsf{BWMP}}) = \sum\limits_{\mathcal{B} \in \mathsf{BSCC}(\mathcal{M})} \mathsf{Pr}(\Diamond \mathcal{B}) \cdot \lambda_{\mathcal{B}}$. Since for each BSCC $\mathcal{B}$, both mean of the minimum mean-cycle in $\mathcal{B}$ and the probability of reaching $\mathcal{B}$ can be computed in polynomial time [15, 16], we obtain the result. ◀

We now consider the direct fixed window mean-payoff function. We show the following.

▶ **Theorem 21.** *Given an MC $\mathcal{M}$, with set S of states, a window length ℓ and a threshold $\lambda \in \mathbb{Q}$, whether $\mathbb{E}^{\mathcal{M}}_{s_{init}}(\mathsf{f}^{\ell}_{\mathsf{DirWMP}}) \geq \lambda$ can be decided in $O(\mathsf{poly}(|S| \cdot \ell \cdot W))$ time.*

We first consider the inductive property of windows (see [8]). For an infinite path $\pi = s_0 \ldots$, a threshold $\lambda \in \mathbb{Q}$, a window length ℓ, a position $i \in \mathbb{N}$ and $l \in [\ell]$, we say that the window starting at position i is *closed* at position $i{+}l$ with respect to λ if $\mathsf{WMP}^{\ell}(\pi(i, \infty)) \geq \lambda$. Otherwise, the window is *open*.

Inductive property of windows. Let $\pi = s_0 \ldots \in \mathsf{Paths}^{\mathcal{M}}$, ℓ be a window length, and λ be a threshold. Assume that a window starting at a position j is open at $j' < j + \ell$ but closed at $j' + 1$. Then, any window starting at a position between j and j' is closed at $j' + 1$.

Note that we cannot focus only on the BSCCs here. Let $\mathsf{f} = \mathsf{f}^{\ell}_{\mathsf{DirWMP}}$. Then, for every path $\pi \in \mathsf{Paths}^{\mathcal{M}}$, we have $\mathsf{f}(\pi) \in \Lambda$ with $\Lambda = \{\frac{p}{q} \mid q \in [\ell],\ p \in [q \cdot W]_0\}$. Let $\Lambda = \{\lambda_0, \ldots, \lambda_n\}$. For every $\lambda_i \in \Lambda$, we construct a new Markov chain $\mathcal{M}^{\lambda_i}_{\ell}$ so that the probability $\mathsf{Pr}(\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, [\lambda_i, \infty[))$ is equal to the probability of not reaching a trap state in $\mathcal{M}^{\lambda_i}_{\ell}$. Thanks to the inductive property of windows, we only need to remember the location of the largest window that is still open, as well as the "amount of payoff" that is required to close it. Hence, in the Markov chain $\mathcal{M}_\ell^{\lambda_i}$, the state space $S' = (S \times [\ell-1]_0 \times [W \cdot (\ell-1)]_0) \cup \{\mathsf{trap}\}$. If the window cannot be closed within ℓ steps, then the state trap is reached. For $\lambda_i \in \Lambda$, we have the following lemma.

▶ **Lemma 22.** $\mathsf{Pr}(\mathsf{f}^{-1}(\mathcal{M}, s_{\mathit{init}}, [\lambda_i, \infty[)) = \mathsf{Pr}(\pi \in \mathsf{Paths}^{\mathcal{M}_\ell^{\lambda_i}} \mid \pi \models \neg\Diamond\{\mathsf{trap}\})$

We can now prove Theorem 21.

Proof sketch of Theorem 21. Assume w.l.o.g. that, in Λ, we have $\lambda_0 < \ldots < \lambda_n$. Now for all $i \leq n-1$, we have $\mathsf{Pr}(\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, \lambda_i)) = \mathsf{Pr}(\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, [\lambda_i, \infty[)) - \mathsf{Pr}(\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, [\lambda_{i+1}, \infty[))$, and $\mathbb{E}_{s_{\mathsf{init}}}^{\mathcal{M}}(\mathsf{f}) = \sum_{i=0}^{n} \mathsf{Pr}(\mathsf{f}^{-1}(\mathcal{M}, s_{\mathsf{init}}, \lambda_i)) \cdot \lambda_i$. Note that $|\Lambda| \leq \ell \cdot W \cdot \ell$ and for each $\lambda_i \in \Lambda$, we have that $|\mathcal{M}_\ell^{\lambda_i}| \leq |\mathcal{M}| \cdot \ell \cdot W \cdot \ell + 1$. Since reachability in Markov chain (here to the trap state) can be decided in polynomial time and W is given in binary, the result follows. ◀

If W is a parameter, we get a fixed parameter tractable algorithm.

References

1 Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT Press, 2008.

2 Benjamin Bordais, Shibashis Guha, and Jean-François Raskin. Expected Window Mean-Payoff. *CoRR*, abs/1812.09298, 2018. arXiv:1812.09298.

3 Tomás Brázdil, Václav Brozek, Krishnendu Chatterjee, Vojtech Forejt, and Antonín Kucera. Two Views on Multiple Mean-Payoff Objectives in Markov Decision Processes. *Logical Methods in Computer Science*, 10(1), 2014.

4 Tomás Brázdil, Krishnendu Chatterjee, Vojtech Forejt, and Antonín Kucera. Trading performance for stability in Markov decision processes. *J. Comput. Syst. Sci.*, 84:144–170, 2017.

5 Tomás Brázdil, Vojtech Forejt, Antonín Kucera, and Petr Novotný. Stability in Graphs and Games. In *27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada*, volume 59 of *LIPIcs*, pages 10:1–10:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

6 Thomas Brihaye, Florent Delgrange, Youssouf Oualhadj, and Mickael Randour. Life is Random, Time is Not: Markov Decision Processes with Window Objectives. *CoRR*, abs/1901.03571, 2019. arXiv:1901.03571.

7 Véronique Bruyère, Quentin Hautem, and Jean-François Raskin. On the Complexity of Heterogeneous Multidimensional Games. In *27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada*, volume 59 of *LIPIcs*, pages 11:1–11:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

8 Krishnendu Chatterjee, Laurent Doyen, Mickael Randour, and Jean-François Raskin. Looking at Mean-Payoff and Total-Payoff through Windows. In *Automated Technology for Verification and Analysis - 11th International Symposium, ATVA 2013, Hanoi, Vietnam, October 15-18, 2013. Proceedings*, volume 8172 of *Lecture Notes in Computer Science*, pages 118–132. Springer, 2013.

9 Krishnendu Chatterjee, Laurent Doyen, Mickael Randour, and Jean-François Raskin. Looking at mean-payoff and total-payoff through windows. *Inf. Comput.*, 242:25–52, 2015.

10 Krishnendu Chatterjee and Monika Henzinger. Efficient and Dynamic Algorithms for Alternating Büchi Games and Maximal End-Component Decomposition. *J. ACM*, 61(3):15:1–15:40, June 2014.

11 Luca De Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, Stanford, CA, USA, 1998.

12 Christoph Haase and Stefan Kiefer. The Odds of Staying on Budget. In *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part II*, pages 234–246, 2015.

13 Paul Hunter, Guillermo A. Pérez, and Jean-François Raskin. Looking at mean payoff through foggy windows. *Acta Inf.*, 55(8):627–647, 2018.

14 Marcin Jurdzinski. Deciding the Winner in Parity Games is in UP ∩ co-UP. *Inf. Process. Lett.*, 68(3):119–124, 1998.

15 Richard M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23:309–311, 1978.

16 Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994.

17 Moshe Y. Vardi. Automatic Verification of Probabilistic Concurrent Finite-State Programs. In *26th Annual Symposium on Foundations of Computer Science, Portland, Oregon, USA, 21-23 October 1985*, pages 327–338. IEEE Computer Society, 1985.

18 Uri Zwick and Mike Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158(1-2):343–359, 1996.