The Complexity of Multi-Mean-Payoff and Multi-Energy Games[⋆,⋆⋆]

Yaron Velner[1], Krishnendu Chatterjee[2], Laurent Doyen[3], Thomas A. Henzinger[2], Alexander Rabinovich[1], and Jean-François Raskin[4]

[1] The Blavatnik School of Computer Science, Tel Aviv University, Israel
[2] IST Austria (Institute of Science and Technology Austria)
[3] LSV, ENS Cachan & CNRS, France
[4] Département d'Informatique, Université Libre de Bruxelles (U.L.B.)

Abstract. In mean-payoff games, the objective of the protagonist is to ensure that the limit average of an infinite sequence of numeric weights is nonnegative. In energy games, the objective is to ensure that the running sum of weights is always nonnegative. Multi-mean-payoff and multi-energy games replace individual weights by tuples, and the limit average (resp. running sum) of each coordinate must be (resp. remain) nonnegative. These games have applications in the synthesis of resource-bounded processes with multiple resources.

We prove the finite-memory determinacy of multi-energy games and show the inter-reducibility of multi-mean-payoff and multi-energy games for finite-memory strategies. We also improve the computational complexity for solving both classes of games with finite-memory strategies: while the previously best known upper bound was EXPSPACE, and no lower bound was known, we give an optimal coNP-complete bound. For memoryless strategies, we show that the problem of deciding the existence of a winning strategy for the protagonist is NP-complete. Finally we present the first solution of multi-mean-payoff games with infinite-memory strategies. We show that multi-mean-payoff games with mean-payoff-sup objectives can be decided in NP $\cap$ coNP, whereas multi-mean-payoff games with mean-payoff-inf objectives are coNP-complete.

Keywords: *Games on graphs; mean-payoff objectives; energy objectives; multi-dimensional objectives.*

1 Introduction

Graph games and multi-objectives. Two-player games on graphs are central in many applications of computer science. For example, in the synthesis problem, implementations of reactive systems are obtained from winning strategies in games with a qualitative objective formalized by an ω-regular specification [22, 21, 1]. In these applications, the games have a qualitative (boolean) objective that determines which player wins. On the other hand, games with quantitative objective which are natural models in economics (where players have to optimize a real-valued payoff) have also been studied in the context of automated design [23, 9, 24]. In the recent past, there has been considerable interest in the design of reactive systems that work in resource-constrained environments (such as

⋆ Preliminary versions appeared in the *Proceedings of the IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science* (FSTTCS), Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, LIPIcs, 2010, pp. 505-516, and in the *Proceedings of the 14th International Conference on Foundations of Software Science and Computational Structures* (FoSSaCS), Lecture Notes in Computer Science 6604, Springer, 2011, pp. 275-289.

⋆⋆ Corresponding author: Laurent Doyen; address: LSV - ENS Cachan, 61 av. du President Wilson, 94235 Cachan Cedex, France; email: doyen@lsv.ens-cachan.fr.

embedded systems). The specifications for such reactive systems are quantitative, and give rise to quantitative games. In most system design problems, there is no unique objective to be optimized, but multiple, potentially conflicting objectives. For example, in designing a computer system, one is interested not only in minimizing the average response time but also the average power consumption. In this work we study such multi-objective generalizations of the two most widely used quantitative objectives in games, namely, *mean-payoff* and *energy* objectives [11, 24, 6, 3].

Multi-mean-payoff games. A *multi-mean-payoff game* is played on a finite weighted game graph by two players. The vertices of the game graph are partitioned into positions that belong to player 1 and positions that belong to player 2. Edges of the graphs are labeled with k-dimensional vectors w of integer values, i.e., $w \in \mathbb{Z}^k$. The game is played as follows. A pebble is placed on a designated initial vertex of the game graph. The game is played in rounds in which the player owning the position where the pebble lies moves the pebble to an adjacent position of the graph using an outgoing edge. The game is played for an infinite number of rounds, resulting in an infinite path through the graph, called a play. The value associated to a play is the mean value in each dimension of the vectors of weights labeling the edges of the play. Accordingly, the winning condition for player 1 is defined by a vector of rational values $v \in \mathbb{Q}^k$ that specifies a threshold for each dimension. A play is winning for player 1 if its vector of mean values is at least v. All other plays are winning for player 2, thus the game is zero-sum. We are interested in the problem of deciding the existence of a winning strategy for player 1 in multi-mean-payoff games. In general infinite memory may be required to win multi-mean-payoff games, but in many practical applications such as the synthesis of reactive systems with multiple resource constraints, the multi-mean-payoff games with finite memory is the relevant problem. Also they provide the framework for the synthesis of specifications defined by mean-payoff conditions [2, 8], and the synthesis question for such specifications under *regular (ultimately periodic)* words correspond to multi-mean-payoff games with finite-memory strategies. Hence we study multi-mean-payoff games both for general strategies as well as finite-memory strategies.

Multi-energy games. In multi-energy games, the winning condition for player 1 requires that, given an initial credit $v_0 \in \mathbb{N}^k$, the sum of v_0 and all the vectors labeling edges up to position i in the play is nonnegative, for all $i \in \mathbb{N}$. The decision problem for multi-energy games asks whether there exists an initial credit v_0 and a strategy for player 1 to maintain the energy nonnegative in all dimensions against all strategies of player 2.

Contributions. In this paper, we study the strategy complexity and computational complexity of solving multi-mean-payoff and multi-energy games. The contributions are as follows.

First, we show that multi-energy and multi-mean-payoff games are determined when played with finite-memory strategies. When considering finite-memory strategies, those games correspond to the synthesis question with ultimately periodic words, and they enjoy pleasant mathematical properties like existence of the limit of the mean value of the weights. We also establish that multi-energy and multi-mean-payoff games are not determined for memoryless strategies. Additionally, we show for multi-energy games determinacy under finite-memory coincides with determinacy under arbitrary strategies, and each player has a winning strategy if and only if he has a finite-memory winning strategy. In contrast, we show for multi-mean-payoff games that determinacy under finite-memory and determinacy under arbitrary strategies do not coincide. Moreover, for multi-mean-payoff games when the strategies for player 1 is restricted to finite-memory strategies, the winning set for player 1 remains unchanged irrespective of whether we consider finite-memory or infinite-memory counter strategies for player 2.

Second, we show that under the hypothesis that both players play either finite-memory or both play memoryless strategies, the decision problems for multi-mean-payoff games and multi-energy games are equivalent.

Third, we study the computational complexity of the decision problems for multi-mean-payoff games and multi-energy games, both for finite-memory strategies and the special case of memoryless strategies. Our complexity results can be summarized as follows. (A) For finite-memory strategies, we provide a nondeterministic polynomial-time algorithm for deciding negative instances of the problems[5]. Thus we show that the decision problems are in coNP. This significantly improves the complexity as compared to the EXPSPACE algorithm that can be obtained by reduction to VASS (vector addition systems with states) [4]. Furthermore, we establish a coNP lower bound for these problems by reduction from the complement of the 3SAT problem, hence showing that the problem is coNP-complete. (B) For the case of memoryless strategies, as the games are not determined, we consider the problem of determining if player 1 has a memoryless winning strategy. First, we show that the problem of determining if player 1 has a memoryless winning strategy is in NP, and then show that the problem is NP-hard even when the weights are restricted to $\{-1, 0, 1\}$ and in dimension 2.

Finally, we study the computational complexity of multi-mean-payoff games for infinite-memory strategies. Our complexity results are summarized as follows. (A) We show that multi-mean-payoff games with mean-payoff-sup objectives can be decided in NP $\cap$ coNP (in the same complexity as for games with single mean-payoff objectives). Moreover, we also show that if mean-payoff games with single mean-payoff objective can be solved in polynomial time, then multi-mean-payoff games with mean-payoff-sup objectives can also be solved in polynomial time. (B) Multi-mean-payoff games with mean-payoff-inf objectives are coNP-complete. (C) Finally, we show that multi-mean-payoff games with combination of mean-payoff-sup and mean-payoff-inf objectives are also coNP-complete.

In summary, our results establish optimal computational complexity results for multi-mean-payoff and multi-energy games under finite-memory, memoryless and infinite-memory strategies.

Related works. Mean-payoff games, which are the one-dimension version of our multi-mean-payoff games, have been extensively studied starting with the works of Ehrenfeucht and Mycielski in [11] where they prove memoryless determinacy for these games. Because of memoryless determinacy, it is easy to show that the decision problem for mean-payoff games lies in NP $\cap$ coNP, but despite large research efforts, no polynomial time algorithm is known for that problem. A pseudo-polynomial time algorithm has been proposed by Zwick and Paterson in [24], and improved in [5]. The one-dimension special case of multi-energy games have been introduced in [6] and further studied in [3] where log-space equivalence with classical mean-payoff games is established.

Multi-energy games can be viewed as games played on VASS (vector addition systems with states) where the objective is to avoid unbounded decreasing of the counters. A solution to such games on VASS is provided in [4] (see in particular Lemma 3.4 in [4]) with a PSPACE algorithm when the weights are $\{-1, 0, 1\}$, leading to an EXPSPACE algorithm when the weights are arbitrary integers. We drastically improve the EXPSPACE upper-bound by providing a coNP algorithm for the problem, and we also provide a coNP lower bound even when the weights are restricted to $\{-1, 0, 1\}$. Finally the work in [12] considers multi-dimension energy games with fixed initial credit, as well as variants of energy games with upper and lower energy bounds.

[5] Negative instances are those where player 1 is losing, and by determinacy under finite-memory where player 2 is winning.

2 Definitions

Well quasi-orders. A relation $\preceq$ over a set D is a *well quasi-order* if the following conditions hold: (a) $\preceq$ is transitive and reflexive, and (b) for all $f : \mathbb{N} \to D$, there exist $i_1, i_2 \in \mathbb{N}$ such that $i_1 < i_2$ and $f(i_1) \preceq f(i_2)$. It is known that $(\mathbb{N}^k, \leq)$ is a well quasi-order and that the Cartesian product of two well quasi-ordered sets is a well quasi-ordered set [10].

Multi-weighted two-player game structures. A *multi-weighted two-player game structure* (or simply a *game*) is a tuple $G = (S_1, S_2, E, w)$ where $S_1 \cap S_2 = \emptyset$, and S_i $(i = 1, 2)$ is the finite set of player-i states (we denote by $S = S_1 \cup S_2$ the state space), $E \subseteq S \times S$ is the set of edges such that for all $s \in S$, there exists $s' \in S$ such that $(s, s') \in E$, and $w : E \to \mathbb{Z}^k$ is the multi-weight labeling function. The parameter $k \in \mathbb{N}$ is the *dimension* of the multi-weights. The game G is a *one-player* game if $S_2 = \emptyset$. The subgraph of G induced by a set $T \subseteq S$ is $G \upharpoonright T = (S_1 \cap T, S_2 \cap T, E \cap (T \times T), w)$. Note that $G \upharpoonright T$ is a game structure if for all $s \in T$, there exists $s' \in T$ such that $(s, s') \in E$.

A *play* in G from an initial state $s_{\mathsf{init}} \in S$ is an infinite sequence $\pi = s_0 s_1 \dots s_n \dots$ of states such that (i) $s_0 = s_{\mathsf{init}}$, and (ii) $(s_i, s_{i+1}) \in E$ for all $i \geq 0$. The *prefix* of length n of π is the finite sequence $\pi(n) = s_0 s_1 \dots s_n$, its last element s_n is denoted $\mathsf{Last}(\pi(n))$ and its length $|\pi(n)|$. The set of all plays in G is denoted $\mathsf{Plays}(G)$.

The *energy level vector* of a play prefix $\rho = s_0 s_1 \dots s_n$ is $\mathsf{EL}(\rho) = \sum_{i=0}^{i=n-1} w(s_i, s_{i+1})$, and the *mean-payoff vectors* of a play $\pi = s_0 s_1 \dots s_n \dots$ are defined as follows (in dimension $1 \leq j \leq k$): $\overline{\mathsf{MP}}(\pi)_j = \limsup_{n \to \infty} \frac{1}{n} \cdot \mathsf{EL}(\pi(n))_j$, and $\underline{\mathsf{MP}}(\pi)_j = \liminf_{n \to \infty} \frac{1}{n} \cdot \mathsf{EL}(\pi(n))_j$.

Strategies. A *strategy* of player i $(i \in \{1, 2\})$ in G is a function $\lambda_i : S^* \cdot S_i \to S$ such that $(s, \lambda_i(\rho \cdot s)) \in E$ for all $\rho \in S^*$ and all $s \in S_i$. A play $\pi = s_0 s_1 \dots \in \mathsf{Plays}(G)$ is *consistent* with a strategy λ_i of player i if $s_{j+1} = \lambda_i(s_0 s_1 \dots s_j)$ for all $j \geq 0$ such that $s_j \in S_i$. The *outcome* from a state s_{init} of a pair of strategies, λ_1 for player 1 and λ_2 for player 2, is the (unique) play from s_{init} that is consistent with both λ_1 and λ_2. We denote $\mathsf{outcome}_G(s_{\mathsf{init}}, \lambda_1, \lambda_2)$ this play. We denote by $T_{\lambda_i(s_{\mathsf{init}})}$ the *strategy tree* obtained as the unfolding of the game G from s_{init} when strategy λ_i is used. The nodes of this tree are all prefixes of the plays from s_{init} that are consistent with the strategy λ_i of player i.

A strategy λ_i for player i uses *finite-memory* if it can be encoded by a deterministic Moore machine $(M, m_0, \alpha_u, \alpha_n)$ where M is a finite set of states (the memory of the strategy), $m_0 \in M$ is the initial memory state, $\alpha_u : M \times S \to M$ is an update function, and $\alpha_n : M \times S_i \to S$ is the next-action function. If the game is in a player-i state $s \in S_i$ and $m \in M$ is the current memory value, then the strategy chooses $s' = \alpha_n(m, s)$ as the next state and the memory is updated to $\alpha_u(m, s)$. Formally, $\langle M, m_0, \alpha_u, \alpha_n \rangle$ defines the strategy λ such that $\lambda(\rho \cdot s) = \alpha_n(\hat{\alpha}_u(m_0, \rho), s)$ for all $\rho \in S^*$ and $s \in S_i$, where $\hat{\alpha}_u$ extends α_u to sequences of states as usual. The strategy is *memoryless* if $|M| = 1$. Given an initial state s_{init} and a finite-memory strategy λ_i of player i, let $G_{\lambda_i(s_{\mathsf{init}})}$ be the graph obtained as the product of G with the Moore machine defining λ_i, with initial vertex $\langle m_0, s_{\mathsf{init}} \rangle$ and where $(\langle m, s \rangle, \langle m', s' \rangle)$ is a transition in the graph if $m' = \alpha_u(m, s)$, and either $s \in S_i$ and $s' = \alpha_n(m, s)$, or $s \in S_{3-i}$ and $(s, s') \in E$.

Objectives. An *objective* for player 1 in G is a set of plays $\varphi \subseteq \mathsf{Plays}(G)$. Given a game G, an initial state s_0, and an objective φ, we say that a strategy λ_1 is *winning* for player 1 from s_0 if for all plays $\pi \in \mathsf{Plays}(G)$ from s_0 that are consistent with λ_1, we have that $\pi \in \varphi$; and we say that a strategy λ_2 is *winning* for player 2 from s_0 if for all plays in $\pi \in \mathsf{Plays}(G)$ from s_0 that are consistent with λ_2, we have that $\pi \notin \varphi$. We denote by $\langle\!\langle 1 \rangle\!\rangle \varphi$ the set of states s_0 such that there exists a winning strategy for player 1 from s_0, and by $\langle\!\langle 2 \rangle\!\rangle \neg\varphi$ the set of states s_0 such that there exists a winning strategy for player 2 from s_0. Note that $\langle\langle 1\rangle\rangle\varphi \cap \langle\langle 2\rangle\rangle\neg\varphi = \emptyset$ by definition. We consider the following objectives:

- *Energy objectives.* Given an initial energy vector $v_0 \in \mathbb{N}^k$, the *multi-energy objective* $\mathsf{PosEnergy}_G(v_0) = \{\pi \in \mathsf{Plays}(G) \mid \forall n \geq 0 : v_0 + \mathsf{EL}(\pi(n)) \geq \{0\}^k\}$ requires that the energy level in all dimensions remain always nonnegative.
- *Mean-payoff objectives.* Given two sets $I, J \subseteq \{1, \ldots, k\}$, the *multi-mean-payoff objective* $\mathsf{MeanPayoffInfSup}_G(I, J) = \{\pi \in \mathsf{Plays}(G) \mid \forall i \in I : \underline{\mathsf{MP}}(\pi)_i \geq 0 \wedge \forall j \in J : \overline{\mathsf{MP}}(\pi)_j \geq 0\}$ requires for all dimensions in I the mean-payoff-inf value be nonnegative, and for all dimensions in J the mean-payoff-sup value be nonnegative.

When the game G is clear from the context we omit the subscript in objective names. Note that arbitrary thresholds $\frac{a}{b} \in \mathbb{Q}$ can be considered in the multi-mean-payoff objectives because the mean-payoff value computed according to the weight function w is greater than $\frac{a}{b}$ if and only if the mean-payoff value according to the weight function $b \cdot w - a$ is greater than 0 where $(b \cdot w - a)(e) = b \cdot w(e) - a$ for all $e \in E$. For the special case of $I = \emptyset$ and $J = \{1, \ldots, k\}$, we denote by $\mathsf{MeanPayoffSup} = \mathsf{MeanPayoffInfSup}(\emptyset, J)$ the conjunction of all mean-payoff-sup objectives, and for $I = \{1, \ldots, k\}$ and $J = \emptyset$ we denote by $\mathsf{MeanPayoffInf} = \mathsf{MeanPayoffInfSup}(I, \emptyset)$ the conjunction of all mean-payoff-inf objectives. We denote by $\mathsf{MeanPayoffSup}_i = \mathsf{MeanPayoffInfSup}(\emptyset, \{i\})$ the single mean-payoff-sup objective in dimension $1 \leq i \leq k$.

Decision problems. We consider the following decision problems:

- The *unknown initial credit problem* asks, given a multi-weighted two-player game structure G, and an initial state s_0, to decide whether there exist an initial credit vector $v_0 \in \mathbb{N}^k$ and a winning strategy λ_1 for player 1 from s_0 for the objective $\mathsf{PosEnergy}_G(v_0)$.
- The *mean-payoff threshold problem* asks, given a multi-weighted two-player game structure G, an initial state s_0, and two sets $I, J \subseteq \{1, \ldots, k\}$ of indices, to decide whether there exists a winning strategy λ_1 for player 1 from s_0 for the objective $\mathsf{MeanPayoffInfSup}_G(I, J)$.

Determinacy, determinacy under finite-memory, and determinacy by finite-memory. We now define the notion of determinacy, determinacy under finite-memory and determinacy by finite-memory.

- *(Determinacy).* A game G with state space S and objective φ is *determined* if from all states $s_0 \in S$, either player 1 or player 2 has a winning strategy, i.e. $S = \langle\langle 1\rangle\rangle\varphi \cup \langle\langle 2\rangle\rangle\neg\varphi$. Observe that since $\langle\langle 1\rangle\rangle\varphi \cap \langle\langle 2\rangle\rangle\neg\varphi = \emptyset$, determinacy means that $\langle\langle 1\rangle\rangle\varphi$ and $\langle\langle 2\rangle\rangle\neg\varphi$ partition the state space.
- *(Determinacy under finite-memory).* We also consider *determinacy under finite-memory* strategies. Let $\langle\langle 1\rangle\rangle^{finite}\varphi$ be the set of states s_0 from which player 1 has a *finite-memory* strategy λ_1 such that for all *finite-memory* strategies λ_2 of player 2, we have $\mathsf{outcome}_G(s_0, \lambda_1, \lambda_2) \in \varphi$. And let $\langle\langle 2\rangle\rangle^{finite}\neg\varphi$ be the set of states s_0 from which player 1 has a *finite-memory* strategy λ_2 such that for all *finite-memory* strategies λ_1 of player 1, we have $\mathsf{outcome}_G(s_0, \lambda_1, \lambda_2) \notin \varphi$. A game G with state space S and objective φ is *determined under finite-memory* if $S = \langle\langle 1\rangle\rangle^{finite}\varphi \cup \langle\langle 2\rangle\rangle^{finite}\neg\varphi$. Again observe that $\langle\langle 1\rangle\rangle^{finite}\varphi \cap \langle\langle 2\rangle\rangle^{finite}\neg\varphi = \emptyset$, and determinacy under finite-memory means that $\langle\langle 1\rangle\rangle^{finite}\varphi$ and $\langle\langle 2\rangle\rangle^{finite}\neg\varphi$ partition the state space. We say that determinacy and determinacy under finite-memory coincide for an objective φ, if for all game structures, we have $\langle\langle 1\rangle\rangle\varphi = \langle\langle 1\rangle\rangle^{finite}\varphi$ and $\langle\langle 2\rangle\rangle\neg\varphi = \langle\langle 2\rangle\rangle^{finite}\neg\varphi$.

– *(Determinacy by finite-memory).* We also consider *determinacy by finite-memory* strategies. Let $\langle\langle 1\rangle\rangle^{fin-inf}\varphi$ be the set of states s_0 from which player 1 has a *finite-memory* strategy λ_1 such that for all strategies λ_2 of player 2, we have $\mathsf{outcome}_G(s_0, \lambda_1, \lambda_2) \in \varphi$ (i.e., player 1 is restricted to finite-memory strategies whereas strategies for player 2 are general infinite-memory strategies). The set of states s_0 from which player 2 has a *finite-memory* strategy λ_2 such that for all strategies λ_1 of player 1, we have $\mathsf{outcome}_G(s_0, \lambda_1, \lambda_2) \notin \varphi$ is denoted $\langle\langle 2\rangle\rangle^{fin-inf}\neg\varphi$. If for all game structures we have $\langle\langle 1\rangle\rangle\varphi = \langle\langle 1\rangle\rangle^{fin-inf}\varphi$ and $\langle\langle 2\rangle\rangle\neg\varphi = \langle\langle 2\rangle\rangle^{fin-inf}\neg\varphi$, and all game structures with objective φ are determined, then we say that determinacy by finite-memory strategies holds for φ.

We first observe that determinacy by finite-memory strategies implies that finite-memory strategies suffice for both players, and determinacy by finite-memory implies determinacy under finite-memory (since given a finite-memory strategy of a player, if there is a counter strategy for the opponent, then there is a finite-memory one by determinacy by finite-memory). Thus determinacy by finite-memory strategies implies that (i) $\langle\langle 1\rangle\rangle\varphi = \langle\langle 1\rangle\rangle^{finite}\varphi = \langle\langle 1\rangle\rangle^{fin-inf}\varphi$; and (ii) $\langle\langle 2\rangle\rangle\neg\varphi = \langle\langle 2\rangle\rangle^{finite}\neg\varphi = \langle\langle 2\rangle\rangle^{fin-inf}\neg\varphi$. As we will show that determinacy and determinacy under finite-memory do not coincide for multi-mean-payoff games (Theorem 5), we consider for multi-mean-payoff objectives φ both (1) winning under finite-memory strategies, i.e. to decide whether $s_0 \in \langle\langle 1\rangle\rangle^{finite}\varphi$ for a given initial state s_0; and (2) winning under general strategies, i.e. to decide whether $s_0 \in \langle\langle 1\rangle\rangle\varphi$ for a given initial state s_0. For multi-energy games we will show determinacy by finite-memory strategies.

Determinacy for multi-mean-payoff and multi-energy objectives follows from a general determinacy result for Borel objectives [19]: (a) multi-mean-payoff objectives can be expressed as a finite intersection of one-dimensional mean-payoff objectives which are complete for the third level of the Borel hierarchy [7]; and (b) multi-energy objectives can be expressed as a finite intersection of one-dimensional energy objectives which are closed sets.

Theorem 1 (Determinacy [19]). *Multi-mean-payoff and multi-energy games are determined.*

Attractors. The player-1 *attractor* of a given set $T \subseteq S$ of target states is the set of states from which player 1 can force to eventually reach a state in T. The attractor is defined inductively as follows: let $A_0 = T$, and for all $j \geq 0$ let

$$A_{j+1} = A_j \cup \{s \in S_1 \mid \exists (s,t) \in E : t \in A_j\} \cup \{s \in S_2 \mid \forall (s,t) \in E : t \in A_j\}$$

denote the set of states from where player 1 can ensure to reach A_j within one step irrespective of the choice of player 2. Then the player-1 attractor is $\mathsf{Attr}_1(T) = \bigcup_{j\geq 0} A_j$. The player-2 *attractor* $\mathsf{Attr}_2(T)$ is defined symmetrically. Note that for $i = 1, 2$, the subgraph $G \restriction (S \setminus \mathsf{Attr}_i(T))$ is again a game structure (i.e., every state has an outgoing edge). For all multi-mean-payoff objectives φ (and in general for all tail objectives [7]), we have $\langle\langle 1\rangle\rangle\varphi = \mathsf{Attr}_1(\langle\langle 1\rangle\rangle\varphi)$ and $\langle\langle 2\rangle\rangle\neg\varphi = \mathsf{Attr}_2(\langle\langle 2\rangle\rangle\neg\varphi)$.

3 Multi-Energy Games

In this section, we study the determinacy and complexity of multi-energy games. First, we show that *finite-memory* strategies are sufficient for player 1, and *memoryless* strategies are sufficient for player 2. It follows that multi-energy games are determined under finite-memory. We establish coNP complexity for the unknown initial credit problem, as well as a matching coNP-hardness result, and we show that under memoryless strategies for player 1 the problem is NP-complete. Finally, we show that the unknown initial credit problem is log-space equivalent to the mean-payoff threshold problem when the players have to use finite-memory strategies (and in general infinite-memory strategies are more powerful than finite-memory strategies in multi-mean-payoff games). The case of infinite-memory strategies in multi-mean-payoff games is addressed in Section 4.

Determinacy under finite-memory. The next lemmas show that finite-memory strategies are sufficient for player 1 in multi-energy games, and that memoryless strategies are sufficient for player 2.

Lemma 1. *For all multi-weighted two-player game structures G and initial states s_0, the answer to the unknown initial credit problem is* Yes *if and only if there exist an initial credit $v_0 \in \mathbb{N}^k$ and a* finite-memory *strategy λ_1^{FM} for player 1 such that for all strategies λ_2 of player 2, $\mathsf{outcome}_G(s_0, \lambda_1^{\mathsf{FM}}, \lambda_2) \in \mathsf{PosEnergy}_G(v_0)$.*

Proof. One direction is trivial. For the other direction, assume that λ_1 is a (not necessary finite-memory) winning strategy for player 1 in G from s_0 with initial credit $v_0 \in \mathbb{N}^k$. We show how to construct from λ_1 a finite-memory strategy λ_1^{FM} that is winning from s_0 against all strategies of player 2 for initial credit v_0.

Consider the strategy tree $T_{\lambda_1(s_0)}$ and associate to each node $\rho = s_0 s_1 \dots s_n$ in this tree the energy vector $v_0 + \mathsf{EL}(\rho)$. Since λ_1 is winning, we have $v_0 + \mathsf{EL}(\rho) \in \mathbb{N}^k$ for all $\rho \in T_{\lambda_1(s_0)}$. Now, consider the relation $\sqsubseteq$ on the set $S \times \mathbb{N}^k$ defined as follows: $(s_1, v_1) \sqsubseteq (s_2, v_2)$ if $s_1 = s_2$ and $v_1 \leq v_2$ (i.e., $v_1(i) \leq v_2(i)$ for all i, $1 \leq i \leq k$). The relation $\sqsubseteq$ is a well quasi-order. As a consequence, on every infinite branch $\pi = s_0 s_1 \dots s_n \dots$ of $T_{\lambda_1(s_0)}$ there exist two indices $i < j$ such that $\mathsf{Last}(\pi(i)) = \mathsf{Last}(\pi(j))$ and $\mathsf{EL}(\pi(i)) \leq \mathsf{EL}(\pi(j))$. We say that node $\pi(j)$ subsumes node $\pi(i)$. Now, let T^{FM} be the tree $T_{\lambda_1(s_0)}$ where we stop each branch when we reach a node n_2 that subsumes one of its ancestor node n_1. By König's lemma [16] and Dickson's lemma [10], the tree T^{FM} is finite. From the node n_2, player 1 can mimic the strategy played in n_1 because the energy level in n_2 is greater than in n_1. From T^{FM}, we can construct the Moore machine of a finite-memory strategy λ_1^{FM} that is winning in the multi-energy game G from s_0 with initial energy level v_0. ☐

Lemma 2 ([4]). *For all multi-weighted two-player game structures G and initial states s_0, the answer to the unknown initial credit problem is* No *if and only if there exists a* memoryless *strategy λ_2 for player 2, such that for all initial credit vectors $v_0 \in \mathbb{N}^k$ and all strategies λ_1 for player 1 we have $\mathsf{outcome}_G(s_0, \lambda_1, \lambda_2) \notin \mathsf{PosEnergy}_G(v_0)$.*

Proof. The proof was given in [4, Lemma 19]. Intuitively, consider a player-2 state $s \in S_2$ with two successors s' and s''. If an initial credit vector v_0' is sufficient for player 1 to win from s_{init} against player 2 always choosing s', and v_0'' is sufficient from s against player 2 always choosing s'', then $v_0' + v_0''$ is sufficient from s_{init} against player 2 arbitrarily alternating between s' and s''. This is because if player 1 maintains the energy nonnegative in all dimensions when the initial credit is v_0, then he can maintain the energy always above Δ when initial credit is $v_0 + \Delta$ ($\Delta \in \mathbb{N}^k$). ☐

The previous two lemmas establishes both determinacy by finite-memory strategies, as well as that determinacy and determinacy under finite-memory coincide. As a consequence of the previous two lemmas, we get the following theorem.

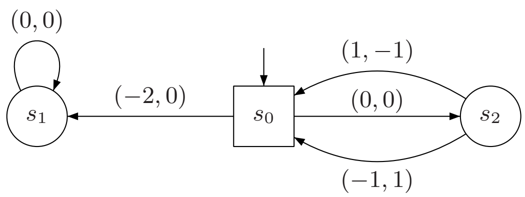

Fig. 1. player 1 (round states) wins with initial credit $(2, 0)$ when player 2 (square states) can use memoryless strategies, but not when player 2 can use arbitrary strategies.

Theorem 2. *Multi-energy games are determined by finite-memory, hence determined under finite-memory. Determinacy coincides with determinacy under finite-memory for multi-energy games.*

Remark 1. Note that even if player 2 can be restricted to play memoryless strategies in multi-energy games, it may be that player 1 is winning with some initial credit vector v_0 when player 2 is memoryless, and is not winning with the same initial credit vector v_0 when player 2 can use arbitrary strategies. This situation is illustrated in Fig. 1 where player 1 (owning round states) can maintain the energy nonnegative in all dimensions with initial credit $(2, 0)$ when player 2 (owning square states) is memoryless. Indeed, either player 2 chooses the left edge from s_0 to s_1 and player 1 wins, or player 2 chooses the right edge from s_0 to s_2, and player 1 wins as well by alternating the edges back to s_0. Now, if player 2 has memory, then player 2 wins by choosing first the right edge to s_2, which forces player 1 to come back to s_0 with multi-weight $(-1, 1)$. The energy level is now $(1, 1)$ in s_0 and player 2 chooses the left edge to s_1 which is losing for player 1. Note that player 1 wins with initial credit $(2, 1)$ and $(3, 0)$ (or any larger credit) against all arbitrary strategies of player 2.

Complexity. We show that the unknown initial credit problem is coNP-complete. First, we show that the one-player version of this game can be solved by checking the existence of a circuit (i.e., a not necessarily simple cycle) with nonnegative effect in all dimensions, and we use the memoryless result for player 2 (Lemma 2) to define a coNP algorithm. Second, we present a coNP-hardness proof.

Theorem 3. *The unknown initial credit problem is coNP-complete.*

First, we need the following result about zero-circuits in multi-weighted directed graphs (a graph is a one-player game). A *zero-circuit* is a finite sequence $s_0 s_1 \dots s_n$ with $n \geq 1$ such that $s_0 = s_n$, $(s_i, s_{i+1}) \in E$ for all $0 \leq i < n$, and $\sum_{i=0}^{n-1} w(s_i, s_{i+1}) = (0, 0, \dots, 0)$. The circuit need not be simple.

Lemma 3 ([18]). *Deciding if a multi-weighted directed graph contains a zero circuit can be done in polynomial time.*

The result of Theorem 3 follows from the next two lemmas.

Lemma 4. *The unknown initial credit problem is in coNP.*

Proof. Let G be a multi-weighted two-player game structure, and s_0 be an initial state. By Lemma 2, we know that player 2 can be restricted to play memoryless strategies. A coNP algorithm guesses a memoryless strategy λ_2 and checks in polynomial time that it is winning for player 2 using the following argument.

Consider the graph $G_{\lambda_2(s_0)}$ as a one-player game (in which all states belong to player 1). We show that if there exists an initial energy level v_0 and an infinite play $\pi = s_0 s_1 \dots s_n \dots$ in $G_{\lambda_2(s_0)}$ such that $\pi \in \mathsf{PosEnergy}(v_0)$, then there exists a reachable circuit in $G_{\lambda_2(s_0)}$ with nonnegative effect in all dimensions. To show this, we extend π with the energy level as follows: let $\pi' = (s_0, w_0)(s_1, w_1) \dots (s_n, w_n) \dots$ where $w_0 = v_0$ and for all $i \geq 1$, $w_i = v_0 + \mathsf{EL}(\pi(i))$. Since $\pi \in \mathsf{PosEnergy}(v_0)$, we know that $w_i \in \mathbb{N}^k$ for all $i \geq 0$. Hence the following order defined on the pairs $(s, w) \in S \times \mathbb{N}^k$ is a well quasi-order: $(s, w) \sqsubseteq (s', w')$ if $s = s'$ and $w(j) \leq w'(j)$ for all $1 \leq j \leq k$. It follows that there exist two indices $i_1 < i_2$ in π' such that $(s_{i_1}, w_{i_1}) \sqsubseteq (s_{i_2}, w_{i_2})$, and the underlying circuit through $s_{i_1} = s_{i_2}$ has nonnegative effect in all dimensions.

Based on this, we can decide if there exists an initial energy vector v_0 and an infinite path in $G_{\lambda_2(s_0)}$ that satisfies $\mathsf{PosEnergy}_G(v_0)$ using the result of Lemma 3 on modified version of $G_{\lambda_2(s_0)}$ obtained as follows. In every state of $G_{\lambda_2(s_0)}$, we add k self-loops with respective multi-weight $(-1, 0, \dots, 0)$, $(0, -1, 0, \dots, 0)$, $\dots$, $(0, \dots, 0, -1)$, i.e. each self-loop removes one unit of energy in one dimension. It is easy to see that $G_{\lambda_2(s_0)}$ has a circuit with nonnegative effect in all dimensions if and only if the modified $G_{\lambda_2(s_0)}$ has a zero circuit, which can be determined in polynomial time. The result follows. □

Lemma 5. *The unknown initial credit problem is coNP-hard.*

Proof. We present a reduction from the complement of the 3SAT problem which is NP-complete [20].

Reduction. We show that the unknown initial credit problem for multi-weighted two-player game structures is at least as hard as deciding whether a 3SAT formula is unsatisfiable. Consider a 3SAT formula ψ in CNF with clauses $C_1, C_2, \dots, C_k$ over variables $\{x_1, x_2, \dots, x_n\}$, where each clause consists of disjunctions of exactly three literals (a literal is a variable or its complement). Given the formula ψ, we construct a game graph as shown in Figure 2. The game graph is as follows: from the initial state, player 1 chooses a clause, then from a clause player 2 chooses a literal that appears in the clause (i.e., makes the clause true). From every literal the next state is the initial state. We now describe the multi-weight labeling function w. In the multi-weight function there is a component for every literal. For edges from the initial state to the clause states, and from the clause states to the literals, the weight for every component is 0. We now define the weight function for the edges from literals back to the initial state: for a literal y, and the edge from y to the initial state, the weight for the component of y is 1, the weight for the component of the complement of y is -1, and for all the other components the weight is 0. We now define a few notations related to assignments of truth values to literals. We consider *assignments* that assign truth values to all the literals. An assignment is *valid* if for every literal the truth value assigned to the literal and its complement are complementary (i.e., for all $1 \leq i \leq n$, if x_i is assigned true (resp. false), then the complement $\overline{x}_i$ of x_i is assigned false (resp. true)). An assignment that is not valid is *conflicting* (i.e., for some $1 \leq i \leq n$, both x_i and $\overline{x}_i$ are assigned the same truth value). If the formula ψ is satisfiable, then there is a valid assignment that satisfies all the clauses. If the formula ψ is not satisfiable, then every assignment that satisfies all the clauses must be conflicting. We now present two directions of the hardness proof.

ψ satisfiable implies player 2 winning. We show that if ψ is satisfiable, then player 2 has a memoryless winning strategy. Since ψ is satisfiable, there is a valid assignment A that satisfies every

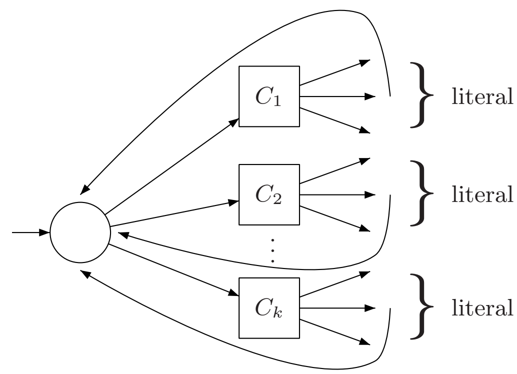

Fig. 2. Game graph construction for a 3SAT formula (Lemma 5).

clause. The memoryless strategy is constructed from the assignment A as follows: for a clause C_i, the strategy chooses a literal as successor that appears in C_i and is set to true by the assignment. Consider an arbitrary strategy for player 1, and the infinite play: the literals visited in the play are all assigned truth values true by A, and the infinite play must visit some literal infinitely often. Consider the literal x that appears infinitely often in the play, then the complement literal $\overline{x}$ is never visited, and every time literal x is visited, the component corresponding to $\overline{x}$ decreases by 1, and since x appears infinitely often it follows that the play is winning for player 2 for every finite initial credit. It follows that the strategy for player 2 is winning, and the answer to the unknown initial credit problem is "No".

ψ not satisfiable implies player 1 is winning. We now show that if ψ is not satisfiable, then player 1 is winning. By determinacy, it suffices to show that player 2 is not winning, and by existence of memoryless winning strategy for player 2 (Lemma 2), it suffices to show that there is no memoryless winning strategy for player 2. Fix an arbitrary memoryless strategy for player 2, (i.e., in every clause player 2 chooses a literal that appears in the clause). If we consider the assignment A obtained from the memoryless strategy, then since ψ is not satisfiable it follows that the assignment A is conflicting. Hence there must exist clause C_i and C_j and variable x_k such that the strategy chooses the literal x_k in C_i and the complement variable $\overline{x}_k$ in C_j. The strategy for player 1 that at the starting state alternates between clause C_i and C_j, along with that the initial credit of 1 for the component of x_k and $\overline{x}_k$, and 0 for all other components, ensures that the strategy for player 2 is not winning. Hence the answer to the unknown initial credit problem is YES, and we have the desired result. □

Observe that our hardness proof works with weights restricted to the set $\{-1, 0, 1\}$. The results of [14] show that in two dimensions ($k = 2$) the unknown initial credit problem with weights in $\{-1, 0, 1\}$ can be solved in polynomial time. The complexity for fixed dimensions $k \geq 3$ is not known. With arbitrary integer weights, the unknown initial credit problem for $k = 1$ is in UP $\cap$ coUP [3].

Complexity for memoryless strategies. We consider multi-energy games when player 1 is restricted to use memoryless strategies. The unknown initial credit problem for memoryless strategies is to decide, given a multi-weighted two-player game structure G, and an initial state s_0, whether there exist an initial credit vector $v_0 \in \mathbb{N}^k$ and a *memoryless* winning strategy λ_1 for player 1 from s_0 for the objective $\mathsf{PosEnergy}_G(v_0)$.

Theorem 4. *The unknown initial credit problem for memoryless strategies is NP-complete.*

Proof. The inclusion in NP is obtained as follows: the polynomial witness is the memoryless strategy for player 1, and once the strategy is fixed we obtain a game graph with choices for player 2 only. The verification is to checks that for every dimension there is no negative cycle, and it can be achieved in polynomial time by solving one-dimensional energy games on graphs with choices for player 2 only [6, 3].

The NP hardness follows from a result of [13] where, given a directed graph and four vertices w, x, y, z, the problem of deciding the existence of two disjoint simple paths (one from w to x and the other from y to z) is shown to be NP-complete. Given such a graph and vertices, construct a one-player game by (1) adding the edges (x, y) with weight $(n, -1)$ and (z, w) with weight $(-1, n)$ (where n is the number of vertices in the graph), and (2) assigning all other edges of the graph the weight $(-1, -1)$. In the resulting one-player game, a winning memoryless strategy from w must induce a simple cycle through w, x, y, z to ensure nonnegative sum of weights in the two dimensions. This show that the unknown initial credit problem for memoryless strategies is at least as hard as the decision problem of [13], and thus NP-hard. The NP-completeness result follows. □

The reduction in the proof of Theorem 4 can be obtained with weights in $\{-1, 0, 1\}$ by replacing the edges with weight n by a sequence of n edges with weight 1. The reduction remains polynomial. Theorem 4 shows NP-hardness for dimension $k = 2$ and weights in $\{-1, 0, 1\}$. For $k = 1$, the problem is solvable in polynomial time with weights in $\{-1, 0, 1\}$, and for arbitrary integer weights, the problem is in UP $\cap$ coUP [3, 5].

Equivalence with multi-mean-payoff games under finite-memory strategies. We show that multi-mean-payoff games where the players are restricted to play finite-memory strategies are log-space equivalent to multi-energy games. The result of Lemma 6 shows that the unknown initial credit problem (for multi-energy games) and the mean-payoff threshold problem (with finite-memory strategies) are equivalent.

Note that if the players use finite-memory strategies, then the outcome π is ultimately periodic (a play $\pi = s_0 s_1 \dots s_n \dots$ is *ultimately periodic* if it can be decomposed as $\pi = \rho_1 \cdot \rho_2^\omega$ where ρ_1 and ρ_2 are two finite sequences of states) and therefore, the value of $\overline{\mathsf{MP}}(\pi)$ and $\underline{\mathsf{MP}}(\pi)$ coincide. We denote by $\mathsf{MeanPayoff}_G$ the set of ultimately periodic plays satisfying the multi-mean-payoff objective $\mathsf{MeanPayoffInf}_G$ (or equivalently, satisfying $\mathsf{MeanPayoffSup}_G$).

Lemma 6. *For all multi-weighted two-player game structures, the answer to the unknown initial credit problem is* Yes *if and only if the answer to the mean-payoff threshold problem under finite-memory strategies is* Yes.

Proof. Let G be multi-weighted two-player game structure of dimension k. First, assume that there exists a winning strategy λ_1 for player 1 in G for the energy objective $\mathsf{PosEnergy}_G(v_0)$ (for some v_0). Theorem 2 establishes that finite memory is sufficient to win multi-energy games, so we can assume that λ_1 has finite memory. Consider the restriction of the graph G_{λ_1} to the reachable vertices, and we show that the energy vector of every simple cycle is nonnegative. By contradiction, if there exists a simple cycle with energy vector negative in one dimension, then the infinite path that reaches this cycle and loops through it forever would violate the objective $\mathsf{PosEnergy}_G(v_0)$ regardless of the vector v_0. Now, this shows that every reachable cycle in G_{λ_1} has nonnegative mean-payoff value in all dimensions, hence λ_1 is winning for the multi-mean-payoff objective $\mathsf{MeanPayoff}_G$.

Second, assume that there exists a finite-memory strategy λ_1 for player 1 that is winning in G for the multi-mean-payoff objective $\mathsf{MeanPayoff}_G$. By the same argument as above, all simple cycles

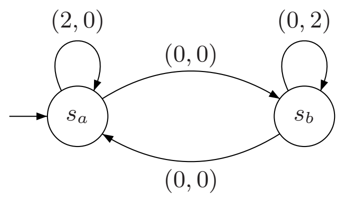

Fig. 3. A multi-mean-payoff game where infinite memory is necessary to win (Lemma 7).

in G_{λ_1} are nonnegative and the strategy λ_1 is also winning for the objective $\mathsf{PosEnergy}_G(v_0)$ for some v_0. Taking $v_0 = \{nW\}^k$ where n is the number of states in G_{λ_1} (which bounds the length of the acyclic paths) and $W \in \mathbb{Z}$ is the largest weight in the game suffices. $\square$

Note that the result of Lemma 6 does not hold for arbitrary strategies as shown in the following lemma.

Lemma 7. *In multi-mean-payoff games, in general infinite-memory strategies are required for winning (i.e., in general, finite-memory strategies are not sufficient for winning).*

Proof. The example of Fig. 3 shows a one-player game. We claim that (a) for $\underline{\mathsf{MP}}$, player 1 can achieve a threshold vector $(1, 1)$, and (b) for $\overline{\mathsf{MP}}$, player 1 can achieve a threshold vector $(2, 2)$; (c) if we restrict player 1 to use a finite-memory strategy, then it is not possible to win the multi-mean-payoff objective with threshold $(1, 1)$ (and thus also not with $(2, 2)$). To prove (a), consider the strategy that visits n times s_a and then n times s_b, and repeats this forever with increasing value of n. This guarantees a mean-payoff vector $(1, 1)$ for $\underline{\mathsf{MP}}$ because in the long-run roughly half of the time is spent in s_a and roughly half of the time in s_b. To prove (b), consider the strategy that alternates visits to s_a and s_b such that after the nth alternation, the self-loop on the visited state s ($s \in \{s_a, s_b\}$) is taken so many times that the average frequency of s gets larger than $\frac{1}{n}$ in the current finite prefix of the play. This is always possible and achieves threshold $(2, 2)$ for $\overline{\mathsf{MP}}$. Note that the above two strategies require infinite memory. To prove (c), recall that finite-memory strategies produce an ultimately periodic play and therefore $\underline{\mathsf{MP}}$ and $\overline{\mathsf{MP}}$ coincide. It is easy to see that such a play cannot achieve $(1, 1)$ because the periodic part would have to visit both s_a and s_b and then the mean-payoff vector (v_1, v_2) of the play would be such that $v_1 + v_2 < 2$ and thus $v_1 = v_2 = 1$ is impossible. $\square$

Lemma 6 and Lemma 7 along with Theorem 2 give the following result.

Theorem 5. *Multi-mean-payoff games are determined under finite-memory, but not determined by finite-memory (i.e., winning strategies in general require infinite-memory, and determinacy and determinacy under finite-memory do not coincide). For multi-mean-payoff objectives φ we have* $\langle\langle 1 \rangle\rangle^{finite}\varphi = \langle\langle 1 \rangle\rangle^{fin-inf}\varphi$.

4 Multi-Mean-Payoff Games

In this section we consider multi-mean-payoff games with infinite-memory strategies (we have already shown in the previous section that multi-mean-payoff games with finite-memory strategies coincide with multi-energy games). We present the following complexity results for the mean-payoff threshold problem: (1) NP $\cap$ coNP for conjunction of $\mathsf{MeanPayoffSup}$ objectives; (2) coNP-completeness for conjunction of $\mathsf{MeanPayoffInf}$ objectives; and (3) coNP-completeness for conjunction of mean-payoff-inf and mean-payoff-sup objectives.

4.1 Conjunction of MeanPayoffSup objectives

We consider multi-weighted two-player game structures with the multi-mean-payoff objective $\mathsf{MeanPayoffSup}_G = \{\pi \in \mathsf{Plays}(G) \mid \overline{\mathsf{MP}}(\pi) \geq (0, 0, \ldots, 0)\})$ for player 1. In general winning strategies for player 1 require infinite memory. We show that memoryless winning strategies exist for player 2 and we present a reduction of the decision problem for a conjunction of k mean-payoff-sup objectives to solving polynomially many instances of the decision problem for single mean-payoff-sup objective. As a consequence the decision problem for $\mathsf{MeanPayoffSup}_G$ lies in NP $\cap$ coNP, and we obtain a pseudo-polynomial time algorithm for this problem.

In the next lemma we show that if player 1 can satisfy the $\mathsf{MeanPayoffSup}$ objective in every individual dimension from all states, then player 1 can satisfy the conjunctive $\mathsf{MeanPayoffSup}$ objective from all states. The converse holds trivially. The main idea of the proof is as follows: for each $1 \leq i \leq k$, let λ_1^i be a winning strategy for player 1 for the objective $\mathsf{MeanPayoffSup}_i$. Intuitively, the winning strategy for the conjunction of mean-payoff-sup objective plays λ_1^i, until the mean-payoff value on dimension i gets larger than a number very close to 0, and then switches to the strategy to $\lambda_1^{(i \pmod{k})+1}$, etc. This way player 1 ensures nonnegative mean-payoff-sup value in every dimension. We present the proof formally below. While memoryless winning strategies exist for each individual dimension, we present a proof that does not use the assumption of witness memoryless winning strategies for individual dimensions. A similar proof technique is used later where memoryless winning strategies for each individual dimension are not guaranteed to exist.

Lemma 8. *If for all states $s \in S$ and for all $1 \leq i \leq k$, player 1 has a winning strategy from s for the objective* $\mathsf{MeanPayoffSup}_i = \{\pi \in \mathsf{Plays} \mid (\overline{\mathsf{MP}}(\pi))_i \geq 0\}$ *(player 1 has winning strategies for each individual dimension), then for all states $s \in S$, player 1 has a winning strategy from s for the objective* $\mathsf{MeanPayoffSup} = \{\pi \in \mathsf{Plays} \mid \overline{\mathsf{MP}}(\pi) \geq (0, 0, \ldots, 0)\}$.

Proof. For each $s \in S$ and $1 \leq i \leq k$, let $\lambda_1^i(s)$ be a winning strategy for player 1 from s for the objective $\mathsf{MeanPayoffSup}_i$, and consider the strategy tree $T_{\lambda_1^i(s)}$. For $\alpha > 0$, we say that a node v of $T_{\lambda_1^i(s)}$ is an α-*good* node if the average of the weights of dimension i of the path from the root to v is at least $-\alpha$. For $Z \in \mathbb{N}$, let $\widehat{T}_\alpha^{i,Z}(s)$ be the tree obtained from $T_{\lambda_1^i(s)}$ by removing all descendants of the α-good nodes that are at depth at least Z. Hence, all branches of $\widehat{T}_\alpha^{i,Z}(s)$ have length at least Z, and the leaves are α-good nodes.

We show that $\widehat{T}_\alpha^{i,Z}(s)$ is a finite tree. By König's Lemma [16], it suffices to show that every path in the tree $\widehat{T}_\alpha^{i,Z}(s)$ is finite. Assume towards contradiction that there is an infinite path π in $\widehat{T}_\alpha^{i,Z}(s)$. Then π is a play consistent with $\lambda_1^i(s)$, and since π does not contain any α-good node beyond depth Z, the mean-payoff-sup value of π in dimension i is at most $-\alpha$, i.e., $(\overline{\mathsf{MP}}(\pi))_i \leq -\alpha$. This contradicts the assumption that $\lambda_1^i(s)$ is a winning strategy for player 1 in dimension i.

We now describe a strategy for player 1 based on the winning strategies of the individual dimensions and show that the strategy is winning for the conjunction of mean-payoff-sup objectives. Let $W \in \mathbb{N}$ be the largest absolute value of the weight function w.

```
1: α ← 1
```

loop

  for $i = 1$ to k **do**

    Let s be the current state, and L be the length of the play so far.

    $Z \leftarrow \frac{L \cdot W}{\alpha}$

    Play according to $\lambda_1^i(s)$ until a leaf of $\widehat{T}_\alpha^{i,Z}(s)$ is reached.

  end for

  $\alpha \leftarrow \frac{\alpha}{2}$

end loop

After the last command in the internal for-loop was executed, the mean-payoff value in dimension i, is at least $\frac{-L \cdot W - m \cdot \alpha}{L+m}$ where $m \geq \frac{L \cdot W}{\alpha}$ and this is at least $\frac{-L \cdot W - m \cdot \alpha}{m} \geq -2 \cdot \alpha$.

Since $\widehat{T}_\alpha^{i,Z}(s)$ is a finite tree, the main loop gets executed infinitely often (i.e., the strategy does not get stuck in the for-loop) and α tends to 0. Thus the supremum of the mean-payoff value is at least 0 in every dimension. Hence the strategy described above is a winning strategy for player 1 for $\mathsf{MeanPayoffSup}$. $\square$

In Lemma 8 the winning strategy constructed for player 1 requires infinite-memory, and by Lemma 7 infinite memory is required in general. For player 2, we show that memoryless winning strategies exist, and we derive the algorithmic solution for the mean-payoff threshold problem.

Lemma 9. *In multi-mean-payoff games with conjunction of* $\mathsf{MeanPayoffSup}$ *objectives for player* 1*, memoryless strategies are sufficient for player* 2*.*

Proof. The proof is by induction on the number of states $|S|$ in the game structure. The base case with $|S| = 1$ is trivial. We now consider the inductive case with $|S| = n \geq 2$. Let $k \in \mathbb{N}$ be the dimension of the weight function w. For $i = 1, \dots, k$, let $W_i = \langle\langle 2 \rangle\rangle \neg \mathsf{MeanPayoffSup}_i$ be the winning region for player 2 for the one-dimensional mean-payoff game played in dimension i. (i.e., in W_i player 2 wins for the objective complementary to $\mathsf{MeanPayoffSup}_i = \{\pi \in \mathsf{Plays} \mid (\overline{\mathsf{MP}}(\pi))_i \geq 0\}$). Let $W = \bigcup_{i=1}^k W_i$. We consider the following two cases:

1. If $W = \emptyset$, then player 1 can satisfy the mean-payoff-sup objective in every dimension, and then by Lemma 8 player 1 wins from everywhere for the objective $\mathsf{MeanPayoffSup} = \{\pi \in \mathsf{Plays} \mid \overline{\mathsf{MP}}(\pi) \geq (0, 0, \dots, 0)\}$. Hence there is no winning strategy for player 2.
2. If $W \neq \emptyset$, then there exists $1 \leq i \leq k$ such that $W_i \neq \emptyset$. In W_i there is a memoryless winning strategy λ_2 for player 2 to falsify $\mathsf{MeanPayoffSup}_i = \{\pi \in \mathsf{Plays} \mid (\overline{\mathsf{MP}}(\pi))_i \geq 0\}$ since memoryless winning strategies exist for both players in mean-payoff games with single objective [11]. The strategy also falsifies $\mathsf{MeanPayoffSup} = \{\pi \in \mathsf{Plays} \mid \overline{\mathsf{MP}}(\pi) \geq (0, 0, \dots, 0)\}$.
   Since W_i is a winning region for player 2, it follows that $W_i = \mathsf{Attr}_2(W_i)$, and the graph G' induced by $S \setminus W_i$ is a game structure. Let $W' = W \setminus W_i$ be the winning region for player 2 in G'. By induction hypothesis (G' has strictly fewer states as a non-empty set W_i is removed), it follows that there is a memoryless winning strategy λ_2' in G' in the region W'. The winning region $S \setminus (W_i \cup W')$ for player 1 in G' is also winning for player 1 in G (since $W_i = \mathsf{Attr}_2(W_i)$, G' is obtained by removing only player 1 edges). Hence to complete the proof it suffices to show that the memoryless strategy obtained by combining λ_2 in W_i and λ_2' in W' is winning for player 2 from $W_i \cup W'$. Define the strategy λ_2^* as follows:

$$\lambda_2^*(s) = \begin{cases} \lambda_2(s) & \text{if } s \in W_i \\ \lambda_2'(s) & \text{if } s \in W'. \end{cases}$$

Consider the memoryless strategy λ_2^* for player 2 and the outcome of any counter strategy for player 1 that starts in $W' \cup W_i$. There are two cases: (a) if the play reaches W_i, then it reaches in finitely many steps, and then λ_2 ensures that player 2 wins; and (b) if the play never reaches W_i, then the play always stays in G', and now the strategy λ_2' ensures winning for player 2. This completes the proof of the second item.

The desired result follows. □

Algorithm. We present Algorithm 1 to solve games with conjunction of mean-payoff-sup objectives. The algorithm maintains the current game structure G_{cur} induced by the current set of states S_{cur}. In every iteration of the repeat-loop, for $i = 1, \ldots, k$, we compute the winning region W_i for player 2 in the current game structure with the single mean-payoff objective on dimension i by a call to $\mathsf{SolveSingleMeanPayoffSup}(G_{cur}, (w)_i)$ which returns the winning region for player 1 in G_{cur} for the objective $\mathsf{MeanPayoffSup}_i$. If W_i is nonempty, then we remove W_i from the current game structure and the iteration continues.

Algorithm 1: SolveMeanPayoffSupGame

Input : A game G with state space S and multi-weight function w.
Output : The winning region of player 1 for objective $\mathsf{MeanPayoffSup} = \bigcap_{1 \le i \le k} \mathsf{MeanPayoffSup}_i$.

begin
    $G_{cur} \leftarrow G$
    $S_{cur} \leftarrow S$
    repeat
        $LosingStatesFound \leftarrow false$
        for $i = 1$ *to* k **do**
            $W_i \leftarrow S_{cur} \setminus \mathsf{SolveSingleMeanPayoffSup}(G_{cur}, (w)_i)$ `/* solves` $\mathsf{MeanPayoffSup}_i$ `*/`
            if $W_i \neq \emptyset$ **then**
                $S_{cur} \leftarrow S_{cur} \setminus W_i$
                $G_{cur} \leftarrow G_{cur} \upharpoonright S_{cur}$
                $LosingStatesFound \leftarrow true$
    until $LosingStatesFound = false$
    return S_{cur}
end

In every iteration the set of states removed from the game structure is certainly winning for player 2. In the end we obtain a game structure such that player 1 wins the mean-payoff objective in every individual dimension from all states, and by Lemma 8 it follows that the remaining region is winning for player 1. Thus game structures with conjunction of mean-payoff-sup objectives can be solved by $O(k \cdot |S|)$ calls to solutions of mean-payoff games with single objective. The following theorem summarizes the results for multi-weighted games with conjunction of mean-payoff-sup objectives.

Theorem 6. *For multi-weighted two-player game structures with objective* $\mathsf{MeanPayoffSup} = \{\pi \in \mathsf{Plays} \mid \overline{\mathsf{MP}}(\pi) \geq (0, 0, \ldots, 0)\}$ *for player 1, the following assertions hold:*

1. *Winning strategies for player 1 require infinite-memory in general, and memoryless winning strategies exist for player 2.*

2. *The problem of deciding whether a given state is winning for player 1 lies in NP $\cap$ coNP.*
3. *The set of winning states for player 1 can be computed with $k \cdot |S|$ calls to a procedure for solving game structures with single mean-payoff objective, hence in pseudo-polynomial time $O(k \cdot |S|^2 \cdot |E| \cdot W)$.*

The results of Theorem 6 are proved as follows. Item 1 follows from Lemma 7 and Lemma 9. Item 3 follows from Algorithm 1 and the results of [5] where an algorithm is given for games with single mean-payoff objectives that works in time $O(|S| \cdot |E| \cdot W)$. We now present the details of Item 2 in two parts. (1) *(In NP).* The NP algorithm guesses the winning region W for player 1, and a memoryless winning strategy λ_1^i for every individual dimension i (such memoryless winning strategies for every individual dimension exist by the results of [11]). The verification procedure checks in polynomial time that for every dimension i the set W is the winning set for player 1 in the graph $G_{\lambda_1^i}$ using the polynomial time algorithm of [15]. The correctness (that is, the existence of winning strategy in every individual dimension implies winning for the conjunction) follows from Lemma 8. (2) *(In coNP).* The coNP algorithm guesses a memoryless winning strategy λ_2 for player 2. The verification procedure needs to solve mean-payoff-sup objectives for the graph G_{λ_2} and by Algorithm 1 this can be solved with $k \cdot |S|$ calls to the polynomial time algorithm of [15] to solve graphs with single mean-payoff objectives. Thus we have the polynomial-time verification procedure, and the coNP complexity bound follows.

4.2 Conjunction of MeanPayoffInf objectives

We consider multi-weighted two-player game structures, and the multi-mean-payoff-inf objective $\mathsf{MeanPayoffInf} = \{\pi \in \mathsf{Plays}(G) \mid \underline{\mathsf{MP}}(\pi) \geq (0, 0, \ldots, 0)\}$ for player 1. In general winning strategies for player 1 require infinite memory (Lemma 7). We show that memoryless winning strategies exist for player 2, and the threshold problem is coNP-complete.

Memoryless strategies for player 2. The objective for player 2 is the complementary objective of player 1. It follows from the results of [17] that memoryless winning strategies exist for player 2 (see Appendix for discussion).

Complexity. We show that the problem of deciding whether a given state is winning for player 1 in multi-weighted game structures with conjunction of mean-payoff-inf objectives is coNP-complete. We first argue about the coNP lower bound.

coNP lower bound. The proof is essentially the same as the proof of Lemma 5 and relies on the existence of memoryless winning strategies for player 2. We consider the hardness proof of Lemma 5 and the reduction used in the lemma. If the formula is satisfiable, then consider the memoryless winning strategy for player 2 constructed from the satisfying assignment. Consider an arbitrary strategy (possibly with infinite-memory) for player 1. Since the strategy for player 2 is constructed from a non-conflicting assignment, it follows that conflicting literals do not appear. Within every three steps some literal is visited. If n is the number of variables, then in any play prefix compatible with the strategy of player 2, the frequency of the literal x with highest frequency in this prefix is at least $\frac{1}{3 \cdot (n+1)}$ (and note that the literal $\overline{x}$ has never appeared). It follows that the average of the weights in the dimension for $\overline{x}$ is at most $-\frac{1}{3 \cdot (n+1)}$ and therefore the mean-payoff-inf objective is violated in some dimension. Conversely, if the formula is not satisfiable, then against every memoryless strategy for player 2, the counter strategy constructed in Lemma 5 (that alternates between the conflicting assignments) ensures that the mean-payoff-inf objective is satisfied. Hence the coNP-hardness follows.

coNP upper bound. The rest of the section is devoted to proving the coNP upper bound. Once a memoryless strategy for player 2 is fixed (as a polynomial witness), we obtain a one-player game structure. To establish the coNP upper bound we need to show that the problem can be solved in polynomial time for one-player game structures. A polynomial-time algorithm for the problem is obtained by solving a variant of the zero circuit problem for multi-weighted directed graphs. The variant of the zero circuit problem is the *nonnegative multi-cycle* problem for directed graphs, where the multi-cycle is not required to be connected by edges as in the case of zero circuit problem.

Nonnegative multi-cycles. Let $G = (V, E, w : E \to \mathbb{Z}^k)$ be a multi-weighted directed graph that is strongly connected. A *multi-cycle* is a multi-set of simple cycles. For a multi-cycle $\mathbf{C}$ we denote by $\mathit{SetCycle}(\mathbf{C})$ the set of cycles that appear in $\mathbf{C}$, and hence $\mathit{SetCycle}(\mathbf{C})$ is a set of simple cycles. For multi-cycle $\mathbf{C} = \{C_1, \ldots, C_n\}$ we denote with m_i the number of occurrences of a simple cycle C_i in the multi-set $\mathbf{C}$, and refer to m_i as the *factor* of C_i. For a simple cycle $C = (e_0, e_1 \ldots e_n)$, we denote $w(C) = \sum_{e \in C} w(e)$. For a multi-cycle $\mathbf{C}$, we denote $w(\mathbf{C}) = \sum_{C \in \mathbf{C}} w(C)$ (note that in the summation a cycle C may appear multiple times in $\mathbf{C}$, and alternatively the summation can be expressed as considering simple cycles C_i that appear in $\mathbf{C}$ and summing up $m_i \cdot w(C_i)$). A *nonnegative multi-cycle* is a non-empty multi-set of simple cycles $\mathbf{C}$ such that $w(\mathbf{C}) \geq 0$ (i.e., in every dimension the weight is nonnegative).

Lemma 10. *Let $G = (V, E, w : E \to \mathbb{Z}^k)$ be a multi-weighted directed graph that is strongly connected.*

1. *The problem of deciding if G has a nonnegative multi-cycle can be solved in polynomial time.*
2. *If G does not have a nonnegative multi-cycle, then there exist a constant $m_G \in \mathbb{N}$ and a real-valued constant $c_G > 0$ such that for all finite paths π^f in the graph G we have $\min\{w_i(\pi^f) \mid i \in \{1, \ldots, k\}\} \leq m_G - c_G \cdot |\pi^f|$.*

Proof. We prove both the items below.

1. The proof of the first item is almost exactly as the proof of Theorem 2.2 in [18]. Given the directed strongly connected graph $G = (V, E, w : E \to \mathbb{Z}^k)$, we consider a variable x_e (for edge coefficient of e) for every $e \in E$. We define the following set of linear constraints.
   (a) For $v \in V$, let $IN(v)$ be the set of all in-edges of v, and $OUT(v)$ be the set of out-edges of v. For every $v \in V$ we define the linear constraint that $\sum_{e \in IN(v)} x_e = \sum_{e \in OUT(v)} x_e$.
   (b) For every $e \in E$ we define the constraint $x_e \geq 0$.
   (c) For every dimension $i \in \{1, \ldots, k\}$, we define the constraint $\sum_{e \in E} x_e \cdot w_i(e) \geq 0$.
   (d) Finally, we define the constraint $\sum_{e \in E} x_e \geq 1$.

   The first set of linear constraints is intuitively the flow constraints; the second constraint specifies that for every edge e, the edge coefficient x_e is nonnegative; the third constraint specifies that in every dimension the sum of edge coefficient time the weights is nonnegative; and the last constraint ensures that at least one edge coefficient is strictly positive (to ensure that the multi cycle is non-empty). This set of constraints can be solved in polynomial time using standard linear programming algorithms. It essentially follows from [18] that this set of linear constraints has a solution iff a nonnegative multi-cycle exists.

2. Let π^f be a finite path in G. The finite path π^f can be decomposed into three paths $\pi_0^f, \pi_c^f, \pi_1^f$ where π_0^f is an initial prefix of length at most $|V|$, π_c^f consists of cycles (not necessarily simple), and π_1^f is a segment of length at most $|V|$ in the end. We can uniquely decompose π_c into a set $\mathbf{C}$ of multi-cycles and hence also into a set of *simple* cycles $\widehat{C} = \mathit{SetCycle}(\mathbf{C}) = \{C_1, \dots, C_n\}$, for $n \leq 2^{|E|}$, such that cycle C_i occurs r_i times in π_c, for some $r_i \in \mathbb{N}$. The sum of the weights in the part of π_c^f is

$$w(\pi_c^f) = \sum_{i=1}^{n} r_i \cdot w(C_i) = (\sum_{i=1}^{n} r_i) \cdot \sum_{i=1}^{n} \frac{r_i}{(\sum_{i=1}^{n} r_i)} \cdot w(C_i) \leq |\pi_c^f| \cdot \sum_{i=1}^{n} \frac{r_i}{(\sum_{i=1}^{n} r_i)} \cdot w(C_i).$$

The second equality is obtained by multiplying and dividing with $(\sum_{i=1}^{n} r_i)$, and the inequality is obtained since $(\sum_{i=1}^{n} r_i) \leq |\pi_c^f|$ (as $|\pi_c^f| = \sum_{i=1}^{n} r_i \cdot |C_i|$). Let $\beta_i = \frac{r_i}{(\sum_{i=1}^{n} r_i)}$ and observe that $\beta_1, \dots, \beta_n \geq 0$ with $\sum_{j=1}^{n} \beta_j = 1$. We first show the existence of a constant $\eta_{\widehat{C}} > 0$, such that for every $\alpha_1, \dots, \alpha_n \geq 0$ with $\sum_{j=1}^{n} \alpha_j = 1$, there exists a dimension $i \in \{1, \dots, k\}$ such that $\sum_{j=1}^{n} \alpha_j \cdot w_i(C_j) \leq -\eta_{\widehat{C}}$.
For every $i \in \{1, \dots, k\}$, we define a function $f_i(\alpha_1, \dots, \alpha_n) = \sum_{j=1}^{n} \alpha_j \cdot w_i(C_j)$ and $f(\alpha_1, \dots, \alpha_n) = \min\{f_i(\alpha_1, \dots, \alpha_n) \mid 1 \leq i \leq k\}$. For every $i \in \{1, \dots, k\}$, the function f_i is continuous. Since f is the minimum of a finite number of continuous functions, f is also continuous. Observe that $[0,1]^n \cap \{(\alpha_1, \dots, \alpha_n) \mid \sum_{j=1}^{n} \alpha_j = 1\}$ is a closed and bounded set. Hence by *Weierstrass theorem* the function f has a maxima c_f in this domain. Let $\alpha_1^*, \dots, \alpha_n^* \geq 0$ such that $f(\alpha_1^*, \dots, \alpha_n^*) = c_f$ and $\sum_{j=1}^{n} \alpha_j^* = 1$. Assume towards contradiction that $c_f \geq 0$, we then show that the linear programming problem on the constraints mentioned above (in item 1) has a solution, which leads to a contradiction. For an edge e, we define the edge coefficient as follows: $x_e = \sum_{e \in C_j \in \widehat{C}} \alpha_j^*$ (i.e., the sum of the α_j^*'s of the cycle the edge belongs to). It follows that all the constraints are satisfied, and this contradicts the assumption that there is no nonnegative multi-cycle. Hence we have $c_f < 0$. Hence it follows that there exists a dimension i such that

$$\begin{aligned} w_i(\pi^f) &\leq (|\pi_0^f| + |\pi_1^f|) \cdot W + c_f \cdot |\pi_c^f| = (|\pi_0^f| + |\pi_1^f|) \cdot W + (|\pi_0^f| + |\pi_1^f|) \cdot (-c_f) + c_f \cdot |\pi^f| \\ &\leq 2 \cdot |V| \cdot (W - c_f) + c_f \cdot |\pi^f|. \end{aligned}$$

Let $m_{\widehat{C}} = \lceil 2 \cdot |V| \cdot (W - c_f) \rceil$ and $\eta_{\widehat{C}} = -c_f$, and we obtain the desired result for the path π^f. Let $\mathcal{C} = \{\mathit{SetCycle}(\mathbf{C}) \mid \mathbf{C} \text{ is a multi-cycle}\}$ be the set of simple cycles of all the multi-cycles of G. Note that $\mathcal{C}$ is a set whose elements are subsets of simple cycles, i.e., $\mathcal{C}$ is the power set of power set of simple cycles and hence $|\mathcal{C}| \leq 2^{2^{|E|}}$. By choosing $m_G = \max_{\widehat{C} \in \mathcal{C}} m_{\widehat{C}}$ and $c_G = \min_{\widehat{C} \in \mathcal{C}} \eta_{\widehat{C}}$ we obtain the desired result.

□

In sequel we abbreviate a maximal strongly connected component of a graph as a scc.

Lemma 11. *Let G be a multi-weighted one-player game structure, and let s_0 be the initial state. If there is a scc C reachable from s_0 such that the multi-weighted directed graph induced by C has a nonnegative multi-cycle, then player 1 has a strategy to satisfy the mean-payoff-inf objective* $\mathsf{MeanPayoffInf}$.

Proof. Let C be a scc reachable from s_0 such that the graph induced by C has a nonnegative multi-cycle. Then there exist simple cycles $C_1, \dots, C_n$, factors $m_1, \dots, m_n$ and finite paths

$\pi_{1,2}, \pi_{2,3}, \ldots, \pi_{n-1,n}, \pi_{n,1}$ such that (i) the path $\pi_{i,j}$ is an acyclic path from C_i to C_j, and (ii) for every $i = 1, \ldots, k$, we have $\sum_{j=1}^{n} m_j \cdot w_i(C_j) \geq 0$. An infinite memory strategy for player 1 is as follows: initialize $Z = 1$, and follow the steps below:

loop
    $Z \cdot m_1$ times in cycle C_1
    $\pi_{1,2}$
    $Z \cdot m_2$ times in cycle C_2
    $\pi_{2,3}$
    $\cdots$
    $Z \cdot m_n$ times in cycle C_n
    $\pi_{n,1}$
    $Z \leftarrow Z + 1$
end loop

Let $L = |\pi_{1,2}| + |\pi_{2,3}| + \cdots |\pi_{n-1,n}| + |\pi_{n,1}|$ be the sum of the lengths of the paths between cycles, and let $P = |C_1| + |C_2| + \cdots + |C_n|$ be the sum of the lengths of the cycles. Note that both L and P are bounded by $2^{|E|} \cdot |S|$ as $n \leq 2^{|E|}$ and each path and cycle is of length at most $|S|$. Consider the steps executed in round $Z + 1$: the sum of weights due to executing the cycles in all previous rounds up to Z is nonnegative in all dimensions. Hence the sum of weights in any dimension, in the steps executed in round $Z + 1$ is at least

$$-(|S| + (Z + 1) \cdot P + Z \cdot L + L) \cdot W.$$

The negative contribution can come from executing the initial prefix of length at most $|S|$ to reach the scc, then the cycles in the present round (bounded by $(Z + 1) \cdot P$ steps) and the paths $\pi_{i,j}$ of length at most L in the previous Z rounds and in the current round (in total bounded by $Z \cdot L + L$ steps). The number of steps executed so far is at least $(L+P) \cdot \sum_{i=1}^{Z} i = (L+P) \cdot \frac{Z \cdot (Z+1)}{2} \geq \frac{(L+P) \cdot Z^2}{2}$. Hence the average for all dimensions for all steps in round $Z + 1$ is at least

$$\begin{aligned} \frac{-2 \cdot (|S| + (Z + 1) \cdot P + Z \cdot L + L) \cdot W}{(L + P) \cdot Z^2} &= \frac{-2 \cdot (|S| + (Z + 1) \cdot (P + L)) \cdot W}{(L + P) \cdot Z^2} \\ &\geq \frac{-2 \cdot |S| \cdot W}{Z^2} + \frac{-2 \cdot (Z + 1) \cdot W}{Z^2}. \end{aligned}$$

As $Z \to \infty$, it follows that the mean-payoff-inf value is at least 0 in every dimension, and hence the result follows. $\square$

Lemma 12. *Let G be a multi-weighted one-player game structure, and let s_0 be the initial state. If for every scc C reachable from s_0 the multi-weighted directed graph induced by C does not have a nonnegative multi-cycle, then player 1 does not have strategy from s_0 to satisfy the mean-payoff-inf objective* $\mathsf{MeanPayoffInf}$.

Proof. Consider an arbitrary strategy for player 1, and let the set of states visited infinitely often be contained in an scc C. Since C does not have a nonnegative multi-cycle it follows from Lemma 10(2) that every infinite path that visits states in C has a mean-payoff-inf value at most $-c$, for some $c > 0$, in some dimension. It follows the strategy for player 1 does not satisfy the mean-payoff-inf objective $\mathsf{MeanPayoffInf}$. $\square$

The following lemma shows that in one-player game structure the $\mathsf{MeanPayoffInf}$ objective can be solved in polynomial time. To describe the precise complexity, let us denote by $\mathsf{LP}(i,j)$ the complexity to solve linear inequalities with i variables and j constraints.

Lemma 13. *Given a multi-weighted one-player game structure G and a state s_0, the problem of deciding whether player 1 has a strategy for a mean-payoff-inf objective* $\mathsf{MeanPayoffInf}$ *from s_0 can be solved in polynomial time (in time $O(|S|+|E|)+\mathsf{LP}(|E|,|S|+|E|+k+1)$).*

Proof. It follows from Lemma 11 and Lemma 12 that an algorithm to solve the problem is as follows: consider the scc decomposition of the graph, and for every multi-weighted graph induced by an scc C reachable from s_0 check if the multi-weighted directed graph induced by C has a nonnegative multi-cycle (in polynomial time by Lemma 10(1)). Since scc decomposition is linear time (in time $O(|S|+|E|)$) and the number of scc's is linear, we obtain the desired result. The complexity of the linear inequations follows from Lemma 10. □

Thus we obtain the desired coNP upper bound. We have the following theorem summarizing the result of this section.

Theorem 7. *For multi-weighted two-player game structures with objective* $\mathsf{MeanPayoffInf} = \{\pi \in \mathsf{Plays} \mid \underline{\mathsf{MP}}(\pi) \geq (0,0,\ldots,0)\}$ *for player 1, the following assertions hold:*

1. *Winning strategies for player 1 require infinite-memory in general, and memoryless winning strategies exist for player 2.*
2. *The problem of deciding whether a given state is winning for player 1 is coNP-complete.*

4.3 Conjunction of MeanPayoffInf and MeanPayoffSup objectives

We consider multi-weighted two-player game structures, two sets $I,J \subseteq \{1,\ldots,k\}$, and the multi-mean-payoff objective $\mathsf{MeanPayoffInfSup}(I,J) = \{\pi \in \mathsf{Plays}(G) \mid \forall i \in I : \underline{\mathsf{MP}}(\pi)_i \geq 0$ and $\forall j \in J : \overline{\mathsf{MP}}(\pi)_j \geq 0\}$ for player 1.

Note that the problem is more general than the problem considered in the previous section (with $J=\emptyset$ we obtain $\mathsf{MeanPayoffInf}$ objectives, and with $I=\emptyset$ we obtain $\mathsf{MeanPayoffSup}$ objectives). Hence it follows that in general winning strategies for player 1 require infinite-memory, and the problem is coNP-hard. We show that memoryless winning strategies exist for player 2, and that the decision problem is coNP-complete.

We start with the crucial result that considers the case when the mean-payoff-sup objective is required for one dimension, and for all the other dimensions the mean-payoff-inf objective is required. The lemma shows that if only one dimension is $\mathsf{MeanPayoffSup}$ objective, then it can be equivalently considered as $\mathsf{MeanPayoffInf}$ objective.

Lemma 14. *Let $I=\{1,\ldots,k-1\}$ and s be a state. Player 1 has a winning strategy for the objective* $\mathsf{MeanPayoffInfSup}(I,\{k\})$ *from s if and only if player 1 has a winning strategy for the objective* $\mathsf{MeanPayoffInf} = \mathsf{MeanPayoffInfSup}(I\cup\{k\},\emptyset)$ *from s.*

Proof. To prove the lemma we show the following equivalent statement: Player 2 has a winning strategy to falsify $\mathsf{MeanPayoffInfSup}(I,\{k\})$ from s if and only if player 2 has a winning strategy to falsify $\mathsf{MeanPayoffInf} = \mathsf{MeanPayoffInfSup}(I\cup\{k\},\emptyset)$ from s.

One direction is trivial as for any sequence $(u_i)_{i\geq 0}$ of real numbers we have $\limsup_{i\to\infty} u_i \geq \liminf_{i\to\infty} u_i$, and hence it follows that a winning strategy for player 2 to falsify $\mathsf{MeanPayoffInfSup}(I, \{i\})$ is also a winning strategy to falsify $\mathsf{MeanPayoffInf}$.

Suppose that player 2 has a winning strategy for $\mathsf{MeanPayoffInf}$, then by Theorem 7 player 2 has a memoryless winning strategy λ_2. Let G_{λ_2} be the one-player game structure obtained by fixing the strategy λ_2 for player 2. Since λ_2 is winning for player 2, it follows from Lemma 11 that in G_{λ_2}, for all scc's C, in the subgraph induced by C there is no nonnegative multi-cycle. It follows from Lemma 10 that there exist a constant $m_{G_{\lambda_2}} \in \mathbb{N}$ and a real-valued constant $c_{G_{\lambda_2}} > 0$ such that for all finite paths π^f in the graph G we have $\min\{w_i(\pi^f) \mid i \in \{1, \ldots, k\}\} \leq m_{G_{\lambda_2}} - c_{G_{\lambda_2}} \cdot |\pi^f|$. Let us denote $c = c_{G_{\lambda_2}}$. We show that λ_2 is winning for player 2 (to falsify $\mathsf{MeanPayoffInfSup}(I, \{k\})$). Consider a play π consistent with λ_2, and assume that $\overline{\mathsf{MP}}(\pi)_k \geq 0$. Then the average payoff in dimension k is greater than $-\frac{c}{2}$ in infinitely many positions (since the limit-superior is at least 0), and by Lemma 10 there is a dimension $1 \leq i < k$ with average payoff at most $-c$ in infinitely many positions, thus $\underline{\mathsf{MP}}(\pi)_i < 0$. Hence either the supremum of the average weight in dimension k is negative, or the infimum of the average weight in one of the other dimensions is negative. In either case, the strategy λ_2 is winning for player 2. This completes the proof. $\square$

Our goal is now to prove a result similar to Lemma 8 for $\mathsf{MeanPayoffInfSup}(I, J)$ objectives. To prove the result, we first prove two lemmas. The following lemma about $\mathsf{MeanPayoffInf}$ objectives is derived from the proof of Lemma 11 and it shows that if player 1 has a winning strategy for a mean-payoff-inf objective (with threshold 0 in every dimension), then for every $\alpha > 0$ there is a finite-memory strategy to ensure mean-payoff-inf value of at least $-\alpha$ in every dimension. Lemma 16 will be a consequence of Lemma 15.

Lemma 15. *Let G be a multi-weighted two-player game structure, and let s_0 be the initial state. If there is a winning strategy for player 1 for the objective* $\mathsf{MeanPayoffInf} = \{\pi \in \mathsf{Plays}(G) \mid \forall 1 \leq i \leq k.\ (\underline{\mathsf{MP}}(\pi))_i \geq 0\}$*, then for all $\alpha > 0$ there is a finite-memory winning strategy for player 1 to ensure the objective* $\mathsf{MeanPayoffInf}(-\alpha) = \{\pi \in \mathsf{Plays}(G) \mid \forall 1 \leq i \leq k.\ (\underline{\mathsf{MP}}(\pi))_i \geq -\alpha\}$.

Proof. Since against finite-memory strategies for player 1 memoryless winning strategies exist for player 2 (Lemma 6 and Lemma 2) and multi-mean-payoff games are determined under finite memory (Theorem 5) to prove that finite-memory winning strategies exist for player 1 for the objective $\mathsf{MeanPayoffInf}(-\alpha)$ we show that against every memoryless strategy for player 2 there exists a finite-memory winning strategy for player 1. Consider a memoryless strategy for player 2 and the one-player game structure obtained after fixing the strategy. By Lemma 12, since player 1 satisfies the $\mathsf{MeanPayoffInf}$ objective, there must be a scc C reachable from s_0 (within $|S|$ steps) such that the graph induced by C has a nonnegative multi-cycle. Then there exist simple cycles $C_1, \ldots, C_n$, factors $m_1, \ldots, m_n$ and finite paths $\pi_{1,2}, \pi_{2,3}, \ldots, \pi_{n-1,n}, \pi_{n,1}$ such that:

1. the path $\pi_{i,j}$ is a path between C_i to C_j with length at most $|S|$.
2. For every $i = 1, \ldots, k$, we have $\sum_{j=1}^{n} m_j \cdot w_i(C_j) \geq 0$

A finite memory strategy for player 1 is as follows: for large enough Z, follow the steps below:

loop
    $Z \cdot m_1$ times in cycle C_1
    $\pi_{1,2}$
    $Z \cdot m_2$ times in cycle C_2

$\pi_{2,3}$
$\cdots$
$Z \cdot m_n$ times in cycle C_n
$\pi_{n,1}$
end loop

In contrast with the strategy of Lemma 11, the above strategy plays the same in every round but for large enough Z, thus it can be implemented with finite memory. Let $L = |\pi_{1,2}| + |\pi_{2,3}| + \cdots |\pi_{n-1,n}| + |\pi_{n,1}|$ be the sum of the lengths of the paths between cycles, and let $M = |C_1| + |C_2| + \cdots + |C_n|$ be the sum of the lengths of the cycles. Note that both L and M are bounded by $2^{|E|} \cdot |S|$ as $n \leq 2^{|E|}$ and each path and cycle is of length at most $|S|$. Consider the steps executed in round i: the sum of weights due to executing the cycles in all previous rounds up to Z is nonnegative in all dimensions. Hence the sum of weights in any dimension, in the steps executed in round i is at least

$$-(|S| + Z \cdot M + i \cdot L + L) \cdot W.$$

The argument is as in Lemma 11. The number of steps executed so far is at least $(L+M) \cdot (i-1) \cdot Z$. Hence the average for all dimensions for all steps in round i is at least

$$-\frac{(|S| + (i+1) \cdot L + Z \cdot M) \cdot W}{(L+M) \cdot (i-1) \cdot Z} \geq -\left(\frac{|S| \cdot W}{Z} + \frac{2 \cdot W}{Z} + \frac{W}{(i-1)} \right),$$

for $i \geq 3$. With Z large enough ($Z \geq \frac{(|S|+2) \cdot W}{\alpha}$), it follows that as $i \to \infty$, the mean-payoff-inf value is at least $-\alpha$ in every dimension, and hence the result follows. $\square$

Lemma 16. *Let G be a multi-weighted two-player game structure, and let s_0 be the initial state. If there is a winning strategy for player 1 for the objective* $\mathsf{MeanPayoffInf} = \{\pi \in \mathsf{Plays}(G) \mid \forall 1 \leq i \leq k.\ (\underline{\mathsf{MP}}(\pi))_i \geq 0\}$*, then for all $\alpha > 0$ there is a finite-memory winning strategy λ and a number N_{α,λ,s_0} such that against all strategies of player 2 and for all $n \in \mathbb{N}$ the sum of weights after n steps is at least $-(N_{\alpha,\lambda,s_0} + n) \cdot \alpha$ in every dimension, i.e., the average of the weights is at least $-2 \cdot \alpha$ once $n \geq N_{\alpha,\lambda,s_0}$.*

Proof. Fix a finite-memory strategy λ for player 1 to satisfy the objective $\mathsf{MeanPayoffInf}(-\alpha) = \{\pi \in \mathsf{Plays}(G) \mid \forall 1 \leq i \leq k.\ (\underline{\mathsf{MP}}(\pi))_i \geq -\alpha\}$ (such a strategy exists by Lemma 15). Let M be the size of the memory. In the game structure obtained by fixing the strategy, in all cycles the average of the weights in every dimension is at least $-\alpha$. For any path it can be decomposed into initial prefix and a cycle free segment in the end (each of length at most $M \cdot |S|$), and the other part is decomposed into cycles (not necessarily simple cycles) (as done in Lemma 10). The initial prefix and trailing prefix is of length at most $M \cdot |S|$ and the sum of the weights is at least $-2 \cdot M \cdot |S| \cdot W$. Hence choosing $N_{\alpha,\lambda,s_0} \geq \frac{2 \cdot M \cdot |S| \cdot W}{\alpha}$ proves the desired result. $\square$

Lemma 17. *Let G be a multi-weighted game structure with multi-mean-payoff objective* $\mathsf{MeanPayoffInfSup}(I, J) = \{\pi \in \mathsf{Plays}(G) \mid \forall i \in I : \underline{\mathsf{MP}}(\pi)_i \geq 0$ *and* $\forall j \in J : \overline{\mathsf{MP}}(\pi)_j \geq 0\}$ *for player 1. For $\ell \in J$, let $\Phi_\ell =$* $\mathsf{MeanPayoffInfSup}(I, \{\ell\})$ *denote the objective that requires to satisfy all* $\mathsf{MeanPayoffInf}$ *objectives and the* $\mathsf{MeanPayoffSup}$ *objective in dimension ℓ. If for all states $s \in S$ and for all $\ell \in J$, player 1 has a winning strategy from s for the objective Φ_ℓ, then for all states $s \in S$, player 1 has a winning strategy from s for the objective* $\mathsf{MeanPayoffInfSup}(I, J)$.

The key idea of the proof is similar to Lemma 8 and we use Lemma 15 (details are presented below for completeness). For all $s \in S$ and all $\ell \in J$, let $\lambda_1^\ell(s)$ be a winning strategy from s for player 1 for the objective Φ_ℓ. Intuitively, the winning strategy for the conjunction of mean-payoff objectives plays $\lambda_1^\ell(\cdot)$ until the mean-payoff value in dimension ℓ gets very close to 0, and then switches to a strategy for another value of $\ell \in J$. Thus player 1 ensures nonnegative mean-payoff value in every dimension, with mean-payoff-inf in dimensions of I and mean-payoff-sup in dimensions of J.

Proof. Let $\alpha > 0$, and s be the initial state. Let $\Phi_\ell(-\frac{\alpha}{2}) = \{\pi \in \mathsf{Plays}(G) \mid \forall i \in I : (\underline{\mathsf{MP}}(\pi))_i \geq -\frac{\alpha}{2}$ and $(\overline{\mathsf{MP}}(\pi))_\ell \geq -\frac{\alpha}{2}\}$. Let $\lambda_{1,\alpha}^\ell(s)$ be a finite-memory winning strategy for player 1 for the objective $\Phi_\ell(-\frac{\alpha}{2})$ with the initial state s (the existence of finite-memory winning strategy for $\Phi_\ell(-\frac{\alpha}{2})$ follows from Lemma 14 and Lemma 15). For $Z \in \mathbb{N}$, consider the tree $\widehat{T}_{\lambda_{1,\alpha}^{\ell,Z}(s)}$ defined as follows. Let $T_{\lambda_{1,\alpha}^\ell(s)}$ be the strategy tree for $\lambda_{1,\alpha}^\ell(s)$ with initial state s. We say that a node v of $T_{\lambda_{1,\alpha}^\ell(s)}$ is an *α-good* node if the average of the weights in all dimensions in I and dimension ℓ of the path from the root to v is at least $-\alpha$. The tree $\widehat{T}_{\lambda_{1,\alpha}^{\ell,Z}(s)}$ is obtained from $T_{\lambda_{1,\alpha}^\ell(s)}$ by removing all descendants of α-good nodes that are at depth at least Z. Hence, the leaves of $\widehat{T}_{\lambda_{1,\alpha}^{\ell,Z}(s)}$ are α-good.

We show that $\widehat{T}_{\lambda_{1,\alpha}^{\ell,Z}(s)}$ is a finite tree. By König's Lemma [16], it suffices to show that every path in the tree is finite. Assume towards contradiction that there is an infinite path π in the tree. Hence π is a play consistent with $\lambda_{1,\alpha}^\ell(s)$, and since π does not contain any α-good node, it follows that for some dimension $i \in I \cup \{\ell\}$ we have $(\overline{\mathsf{MP}}(\pi))_i \leq -\alpha$ (and $(\underline{\mathsf{MP}}(\pi))_i \leq -\alpha$ as well). It follows that $\pi \notin \Phi_\ell(-\frac{\alpha}{2})$. This contradicts the assumption that $\lambda_{1,\alpha}^\ell(s)$ is a winning strategy for player 1 for $\Phi_\ell(-\frac{\alpha}{2})$.

We now describe a strategy for player 1 based on the finite-memory winning strategies for $\Phi_\ell(-\frac{\alpha}{2})$ and show that the strategy is winning for the objective $\mathsf{MeanPayoffInfSup}(I, J)$.

```
α ← 1
loop
   for ℓ ∈ J do
      Let s be the current state, and L be the play length so far.
      Z ← max{L·W/α, N*_{α/2}} (where N*_{α/2} = max{N_{α/2, λ̂(s), s} | s ∈ S, ℓ' ∈ J, λ̂(s) = λ^{ℓ'}_{1,α/2}(s)}, that is,
          λ̂(s) = λ^{ℓ'}_{1,α/2}(s) is the finite-memory strategy for Φ_{ℓ'}(−α/2) from s, the number N_{α/2, λ̂(s), s} is
          as defined in Lemma 16 for the strategy, and N*_{α/2} is the maximum over ℓ' ∈ J)
      Play according to λ^ℓ_{1,α}(s) until a leaf s' of T̂_{λ^{ℓ,Z}_{1,α}(s)} is reached.
   end for
   α ← α/2
end loop
```

Let $W \in \mathbb{N}$ be the largest absolute value of the weight function w. After the last command in the internal for-loop was executed, the mean-payoff value in dimension ℓ, is at least $\frac{-L \cdot W - Z \cdot \alpha}{L+Z}$ where $Z \geq \frac{L \cdot W}{\alpha}$ and this is at least

$$\frac{-L \cdot W - \alpha \cdot \frac{L \cdot W}{\alpha}}{L + \frac{L \cdot W}{\alpha}} \geq -2 \cdot \alpha.$$

Consider the segment of the play for the round for a value of α: let us denote by M_b the number of steps played till the beginning of the round and we will denote by M_t the total number of steps of the current round. Our goal is to obtain an upper bound on the average of the weights for all $n \leq M_b + M_t$. In the beginning of the round (i.e., after M_b steps) the average value for dimensions in I is at least $-2 \cdot \alpha$ (recall that α has been halved in line 8). Step 5 ensures that at least N_α^* steps have been already played, i.e., $M_b \geq N_\alpha^*$. It follows from Lemma 16 that for all dimensions in I and for all steps $M_b \leq n \leq M_b + M_t$ of the current round, the sum of the weights is at least $-(M_b \cdot 2 \cdot \alpha + N_\alpha^* \cdot \alpha + (n - M_b) \cdot \alpha)$, and hence the average value at step n is at least

$$\frac{-(M_b \cdot 2 \cdot \alpha + N_\alpha^* \cdot \alpha + (n - M_b) \cdot \alpha)}{n} \geq -4 \cdot \alpha$$

since $n \geq M_b$ and $n \geq N_\alpha^*$. That is, for all steps in the round for α, for all dimensions in I, the average value is at least $-4 \cdot \alpha$. In every external for-loop α gets smaller, and L gets bigger. Moreover, since the tree $\widehat{T}_{\lambda_{1,\alpha}^{\ell,Z}(s)}$ is finite, it follows that the main loop gets executed infinitely often (i.e., the strategy does not get stuck in the for-loop). Thus when the length of the play tends to infinity, the supremum of the mean-payoff value tends to a value at least 0 in every dimension $j \in J$, and the infimum of the mean-payoff value tends to a value at least 0 in every dimension $i \in I$. Hence the strategy described above is a winning strategy for player 1. □

Lemma 18. *In multi-mean-payoff games with objective* $\mathsf{MeanPayoffInfSup}(I, J)$ *for player* 1*, memoryless strategies are sufficient for player* 2*.*

Proof. The proof is similar to the proof of Lemma 9, and based on induction on the number of states $|S|$ in the game structure. The base case with $|S| = 1$ is obvious. We now consider the inductive case with $|S| = n \geq 2$. For $\ell \in J$, let W_ℓ be the winning region for player 2 for the objective Φ_ℓ as defined in Lemma 17. Let $W = \bigcup_{\ell \in J} W_\ell$. We consider the following two cases:

1. If $W = \emptyset$, then player 1 can satisfy the objective Φ_ℓ for all $\ell \in J$, and then by Lemma 17 player 1 wins from everywhere for the objective $\mathsf{MeanPayoffInfSup}(I, J)$. Hence there is no winning strategy for player 2.
2. If $W \neq \emptyset$, then there exists $\ell \in J$ such that $W_\ell \neq \emptyset$. In W_ℓ there is a memoryless winning strategy λ_2 for player 2 to falsify Φ_ℓ, and the strategy also falsifies $\mathsf{MeanPayoffInfSup}(I, J)$ as $\mathsf{MeanPayoffInfSup}(I, J) = \bigcap_{\ell \in J} \Phi_\ell$. The existence of memoryless winning strategy for player 2 follows from the following facts: by Lemma 14 it follows that if player 2 can falsify the objective Φ_ℓ, then player 2 can also falsify the objective where in the dimension ℓ we consider the mean-payoff-inf objective instead of mean-payoff-sup objective, and the existence of memoryless strategies against mean-payoff-inf objectives follows from Theorem 7. The rest of the proof is identical to the proof of Lemma 9 and can be omitted (we present it for sake of completeness). Since W_ℓ is a winning region for player 2 it follows that $W_\ell = \mathsf{Attr}_2(W_\ell)$, and hence the graph G' induced by $S \setminus W_\ell$ is a game structure. Let $W' = W \setminus W_\ell$ be the winning region for player 2 in G'. By inductive hypothesis (since G' has strictly fewer states as a non-empty set W_ℓ is removed), it follows that there is a memoryless winning strategy λ_2' in G' for the region W'. The winning region $S \setminus (W_\ell \cup W')$ for player 1 in G' is also winning for player 1 in G (since $W_\ell = \mathsf{Attr}_2(W_\ell)$, G' is obtained by removing only player 1 edges). Hence to complete the proof it suffices to show that the memoryless strategy obtained by combining λ_2 in W_ℓ and λ_2' in W'

is winning for player 2 from $W_\ell \cup W'$. Define the strategy λ_2^* as follows:

$$\lambda_2^*(s) = \begin{cases} \lambda_2(s) & \text{if } s \in W_\ell \\ \lambda_2'(s) & \text{if } s \in W'. \end{cases}$$

Consider the memoryless strategy λ_2^* for player 2 and the outcome of any counter strategy for player 1 that starts in $W' \cup W_\ell$. There are two cases: (a) if the play reaches W_ℓ, then it reaches in finitely many steps, and then λ_2 ensures that player 2 wins; and (b) if the play never reaches W_i, then the play always stays in G', and now the strategy λ_2' ensures winning for player 2.

The desired result follows. □

coNP upper bound. Since memoryless winning strategies exist for player 2, to establish the coNP upper bound we need to show that one-player game structures with $\mathsf{MeanPayoffInfSup}(I,J)$ objectives can be solved in polynomial time. First we interpret $\mathsf{MeanPayoffInfSup}(I,J)$ as the conjunction of Φ_ℓ for $\ell \in J$. From Lemma 14 it follows every Φ_ℓ can be considered as $\mathsf{MeanPayoffInf}$ objective and hence can be solved in polynomial time for one-player game structures by the results of Section 4.2. Hence the coNP upper bound follows. We have the following theorem summarizing the results of this section.

Theorem 8. *For multi-weighted two-player game structures with objective* $\mathsf{MeanPayoffInfSup}(I,J) = \{\pi \in \mathsf{Plays}(G) \mid \forall i \in I : \underline{\mathsf{MP}}(\pi)_i \geq 0 \textit{ and } \forall j \in J : \overline{\mathsf{MP}}(\pi)_j \geq 0\}$ *for player 1, the following assertions hold:*

1. *Winning strategies for player 1 require infinite-memory in general, and memoryless winning strategies exist for player 2.*
2. *The problem of deciding whether a given state is winning for player 1 is coNP-complete.*

5 Conclusion

In this work we considered games with multiple mean-payoff and energy objectives, and established determinacy under finite-memory, inter-reducibility of these two classes of games for finite-memory strategies, and improved the complexity bounds from EXPSPACE to coNP-complete. We also showed that multi-energy and multi-mean-payoff games under memoryless strategies are NP-complete. Finally, we studied multi-mean-payoff games with infinite-memory strategies and show that multi-mean-payoff games with mean-payoff-sup objectives can be decided in NP $\cap$ coNP (and can be solved in polynomial time if mean-payoff games with single objective can be solved in polynomial time); and multi-mean-payoff games with mean-payoff-inf objectives, and combination of mean-payoff-inf and mean-payoff-sup objectives are coNP-complete. Thus we present optimal computational complexity results for multi-energy and multi-mean-payoff games under finite-memory, memoryless, and infinite-memory strategies.

Acknowledgement. We are grateful to Jean Cardinal for pointing the reference [18].

References

1. M. Abadi, L. Lamport, and P. Wolper. Realizable and unrealizable specifications of reactive systems. In *Proc. of ICALP*, LNCS 372, pages 1–17. Springer, 1989.

2. R. Alur, A. Degorre, O. Maler, and G. Weiss. On omega-languages defined by mean-payoff conditions. In *Proc. of FOSSACS*, LNCS 5504, pages 333–347. Springer, 2009.
3. P. Bouyer, U. Fahrenberg, K. G. Larsen, N. Markey, and J. Srba. Infinite runs in weighted timed automata with energy constraints. In *Proc. of FORMATS*, LNCS 5215, pages 33–47. Springer, 2008.
4. T. Brázdil, P. Jancar, and A. Kucera. Reachability games on extended vector addition systems with states. In *Proc. of ICALP*, LNCS 6199, pages 478–489. Springer, 2010.
5. L. Brim, J. Chaloupka, L. Doyen, R. Gentilini, and J.-F. Raskin. Faster algorithms for mean-payoff games. *Formal Methods in System Design*, 38(2):97–118, 2011.
6. A. Chakrabarti, L. de Alfaro, T. A. Henzinger, and M. Stoelinga. Resource interfaces. In *Proc. of EMSOFT: Embedded Software*, LNCS 2855, pages 117–133. Springer, 2003.
7. K. Chatterjee. Concurrent games with tail objectives. *Theor. Comput. Sci.*, 388(1-3):181–198, 2007.
8. K. Chatterjee, L. Doyen, H. Edelsbrunner, T. A. Henzinger, and P. Rannou. Mean-payoff automaton expressions. In *Proc. of CONCUR*, LNCS 6269, pages 269–283. Springer, 2010.
9. A. Condon. The complexity of stochastic games. *Inf. Comput.*, 96(2):203–224, 1992.
10. L. E. Dickson. Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors. *Am. J. of Mathematics*, 35(4):413–422, 1913.
11. A. Ehrenfeucht and J. Mycielski. Positional strategies for mean-payoff games. *Int. J. of Game Theory*, 8:109–113, 1979.
12. U. Fahrenberg, L. Juhl, K. G. Larsen, and J. Srba. Energy games in multiweighted automata. In *Proc. of ICTAC: Theoretical Aspects of Computing*, LNCS 6916, pages 95–115. Springer, 2011.
13. S. Fortune, J. E. Hopcroft, and J. Wyllie. The directed subgraph homeomorphism problem. *Theor. Comput. Sci.*, 10:111–121, 1980.
14. Chaloupka J. Z-reachability problem for games on 2-dimensional vector addition systems with states is in P. In *Proceedings of RP 2010: Reachability Problems*, LNCS 6227, pages 104–119. Springer-Verlag, 2010.
15. R.M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23:309–311, 1978.
16. D. König. *Theorie der endlichen und unendlichen Graphen*. Akademische Verlagsgesellschaft, Leipzig, 1936.
17. E. Kopczynski. Half-positional determinacy of infinite games. In *ICALP (2)*, pages 336–347, 2006.
18. S. R. Kosaraju and G. F. Sullivan. Detecting cycles in dynamic graphs in polynomial time (preliminary version). In *Proc. of STOC: Symposium on Theory of Computing*, pages 398–406. ACM, 1988.
19. D. Martin. Borel determinacy. In *Annals of Mathematics*, volume 102, pages 363–371, 1975.
20. C.H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1993.
21. A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *Proc. of POPL*, pages 179–190, 1989.
22. P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete event processes. *SIAM Journal on Control and Optimization*, 25(1):206–230, 1987.
23. L. S. Shapley. Stochastic games. In *Proc. of the National Acadamy of Science USA*, volume 39, pages 1095–1100, 1953.
24. U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Th. Comp. Sc.*, 158:343–359, 1996.

Appendix

We discuss the results of [17] which shows the existence of memoryless winning strategies for player 2 when the objective for player 1 is the conjunction of mean-payoff-inf objectives. We will also argue that the results of [17] do not show the existence of memoryless winning strategies for player 2 when the objective for player 1 is the conjunction of mean-payoff-sup objectives (the result that we establish in Lemma 9). The result of [17] requires the notion of *convexity* for *prefix-independent* objectives.

Prefix-independent and convex objectives. An objective φ is prefix-independent if for all plays π and π' such that $\pi' = \rho \cdot \pi$, where ρ is a finite prefix, we have $\pi \in \varphi$ iff $\pi' \in \varphi$, i.e., the objective is independent of finite prefixes. A play π is a *combination* of two plays $\pi_1 = u_1 u_3 u_5 \dots$ and $\pi_2 = u_0 u_2 u_4 \dots$, where u_i's are finite prefixes, if $\pi = u_0 u_1 u_2 u_3 u_4 \dots$. An objective φ is *convex* if it is closed under combination. We refer the reader to [17] for further details. The results of [17] shows that if the objective for player 1 is prefix-independent and convex, then memoryless winning strategies exist for player 2. It is easy to verify that mean-payoff-inf objectives are both prefix-independent and convex. It follows that conjunction of mean-payoff-inf objectives are also prefix-independent and convex. Hence in games with conjunction of mean-payoff-inf objectives, memoryless winning strategies exist for player 2. We now show with an example that in contrast mean-payoff-sup objectives are not convex.

Example 1. Consider a one-player game structure G with two states $\{s_+, s_-\}$, with all edges, such that all incoming edges to state s_+ have weight $+2$, and all incoming edges to s_- have weight -2.

Consider the following play π_0:

1. *Step 1.* Repeat the self-loop in s_- until the average weight of the play prefix is below -1, then take edge to $s+$ and goto Step 2.
2. *Step 2.* Repeat the self-loop in s_+ until the average weight of the play prefix is above 1, then take edge to s_- and goto Step 1.

Consider the play π_1 obtained by exchanging s_+ and s_- in π_0. It is easy to verify that $\overline{\mathsf{MP}}(\pi_0) = \overline{\mathsf{MP}}(\pi_1) = +1$. However, for the following combination of the plays π_2, such that forall $i \geq 0$ the $2i-1$-th state of π_2 is the i-th state of π_0 and the $2i$-th state of π_2 is the i-th state of π_1. We get that $\overline{\mathsf{MP}}(\pi_2) = 0$. It follows that mean-payoff-sup objectives are not convex.