# Fair Quantitative Games

Ashwani Anand [0000–0002–9462–8514], Satya Prakash Nayak [0000–0002–4407–8681], Ritam Raha [0000–0003–1467–1182], Irmak Sağlam [0000–0002–4757–1631], and Anne-Kathrin Schmuck [0000–0003–2801–639X]

Max Planck Institute for Software Systems, Kaiserslautern, Germany

Abstract. We examine two-player games over finite weighted graphs with quantitative (mean-payoff or energy) objective, where one of the players additionally needs to satisfy a fairness objective. The specific fairness we consider is called strong transition fairness, given by a subset of edges of one of the players, which asks the player to take fair edges infinitely often if their source nodes are visited infinitely often. We show that when fairness is imposed on player 1, these games fall within the class of previously studied  $\omega$ -regular mean-payoff and energy games. On the other hand, when the fairness is on player 2, to the best of our knowledge, these games have not been previously studied. We provide gadgetbased algorithms for fair mean-payoff games where fairness is imposed on either player, and for fair energy games where the fairness is imposed on player 1. For all variants of fair mean-payoff and fair energy (under unknown initial credit) games, we give pseudo-polynomial algorithms to compute the winning regions of both players. Additionally, we analyze the strategy complexities required for these games. Our work is the first to extend the study of strong transition fairness, as well as gadget-based approaches, to the quantitative setting. We thereby demonstrate that the simplicity of strong transition fairness, as well as the applicability of gadget-based techniques, can be leveraged beyond the  $\omega$ -regular domain.

### 1 Introduction

Games on graphs serve as a formal and effective framework for automatically synthesizing correct-by-design software in cyber-physical systems (CPS). In this setting, one player represents a controller aiming to ensure a high-level logical specification in response to the actions of an adversarial player representing the external environment. Two-player graph games abstract the strategic reactive behavior of autonomous systems, such as robots [36,45,52] or cars [39,40], and can be employed to synthesize *correct-by-design* strategies through the algorithmic process of reactive synthesis. Such strategies function as logical controllers to ensure the required behavioral guarantees, ensuring the system meets its safety and performance goals over its strategic interactions with its environment.

In the CPS context, these game-based abstractions have been augmented with additional information, such as assumptions about the players' strategic behavior induced by the underlying physical processes [5,33], and quantitative objectives that allow to optimize strategies w.r.t. given performance metrics [8].

These enhancements enable reactive synthesis algorithms to compute control strategies that are more aligned with the specific requirements of CPS applications. However, these extensions typically come with increased computational costs, hindering their practical applicability. In this paper, we show that a particular subclass of such games – namely, energy and mean-payoff games under strong-transition-fairness constraints – distinctly posses favorable computational properties. We refer to these games as fair energy and fair mean-payoff games.

## 1.1 Background

Strong Transition Fairness. Strong transition fairness [9,33,43] is defined over specific edges, referred to as fair edges, and restricts strategies to use a fair edge infinitely often if its source vertex is visited infinitely often. Strong transition fairness constraints are less expressive than general fairness, typically expressed via a qualitative objective called the Streett objective [47]. Yet, they naturally arise in areas such as resource management [18], abstractions of continuous-time physical processes for planning [6, 25, 30] and controller synthesis [37, 41, 49]. At the same time, games with strong transition fairness conditions offer more favorable computational properties compared to those with Streett fairness, making them a compelling subject of study.

Despite their strong motivation from CPS applications and their favorable computational properties, strong transition fairness has so far been considered only in the context of qualitative objectives, such as parity [34,44] or Rabin [10,48], where it is shown to be computationally inexpensive, preserving the overall complexity of the game and inducing negligible computational cost. This paper shows that these property carry over to quantitative games.

Energy & Mean-Payoff Games. Quantitative objectives, such as energy or mean-payoff, allow capturing strategic limitations induced by constrained resources, such as the power usage of embedded components, the buffer size of a networking element [27], or string selections with limited storage [53].

Mean-payoff games, introduced in [32], assign integer weights to edges, representing the payoffs received by player 1 (the controller) and paid by player 2 (the environment) when an edge is taken. These games are played over infinitely many rounds, with player 1 aiming to maximize the long run average value of the edges traversed, while player 2 aims to minimize this value.

Energy games, introduced more recently in [17], also assign integer weights to edges, this time representing energy gains or losses. In this setting, the controller's goal is to construct an infinite path where the total energy in every prefix remains non-negative, while the environment aims to prevent this.

It is known that determining the winner in both energy and mean-payoff games is in NP  $\cap$  coNP [54]. The state-of-the-art algorithms for mean-payoff games have runtime  $\mathcal{O}(n^2mW)$  (and a runtime of  $\mathcal{O}(nmW)$  for the threshold problem), where n is the number of nodes, m is the number of edges and W is the maximum absolute weight in the game arena [28]. For energy games, the best algorithms are either deterministic with the run time  $\mathcal{O}(\min(mnW, mn2^{n/2} \log W))$  [15, 31] or randomized with the run time  $2^{\mathcal{O}(\sqrt{n \log n})}$  [11].

Combining Quantitative & Qualitative Objectives. Conceptually, fair energy and fair mean-payoff games considered in this paper combine quantitative (i.e., energy or mean-payoff) with qualitative (i.e. strong transition fairness) obligations to constrain the moves of the players in the resulting game.

Combinations of quantitative and qualitative objectives have been studied for several variants of mean-payoff and energy objectives combined with general  $\omega$ -regular goals [12, 17, 20, 23, 29, 35]. In particular, energy parity games were introduced in [20], demonstrating these games' inclusion in NP  $\cap$  coNP and polynomial equivalence to mean-payoff parity games, previously studied in [23]. Further, people have studied the combination of quantitative objectives (optimization or threshold variant) with (not necessarily  $\omega$ -regular) Boolean constraints [1–4, 13, 14, 16, 21, 22, 26, 51]. One example is the Beyond Worst-Case synthesis framework, introduced by Bruyère et al. [16], which combines Boolean guarantees (mean-payoff thresholds) with quantitative performance optimization (expected payoff).

It is known that strong transition fairness conditions can be translated into a classical Streett condition, i.e., a subclass of (qualitative)  $\omega$ -regular objectives, by transforming each fair edge e=(u,v) into a Streett pair ( $\{u\},\{e\}$ ) in the original game arena. If only player 1 (the controller) is constrained by the fairness, this transformation results in an energy (resp. mean-payoff) Streett game. Following [7], Energy Streett games<sup>1</sup> can be solved in one of the following ways:

- Encode the Streett energy objective as a  $\mu$ -calculus formula and solve the formula using symbolic fixed-point computations: The computation takes  $\mathcal{O}(n \cdot [b+1]^{\left\lfloor \frac{d}{2}+1 \right\rfloor})$  symbolic steps, where  $b=(d+1)\cdot \left(((n^2+n)\frac{n}{2}!-1)W\right)$  is an upper bound on the credit a Streett energy game requires, where d is the number of Streett pairs, which in our case corresponds to the number of fair edges. This complexity can be simplified to  $\mathcal{O}\left(\left(\frac{n}{2}!\cdot W\right)^{\frac{m}{2}}\right)$  by omitting the polynomial factors where m is the number of edges; which is super-exponential.
- ▶ Translate the energy Streett game into a Streett game by encoding the domain of energy levels in the state space. This explodes the state space by multiplying the number of states with the above-mentioned upper-bound b. Using state-of-the-art algorithms for Streett games [42], this gives an algorithm with worst-case super-exponential runtime of  $\mathcal{O}((\frac{n}{2}!W)^{m+1}m!)$ .

If, however, player 2 (the environment) is constrained by strong-transition fairness, the above reduction to energy (resp. mean-payoff) Streett games does not apply as the energy or mean-payoff objectives are not symmetric for both players. In this case, the given energy (resp. mean payoff) game is enhanced with a fairness assumption weakening the opponent. While such games have been investigated for qualitative objectives [34], this paper is the first to consider quantitative games under such fairness assumptions.

<sup>&</sup>lt;sup>1</sup> This also gives a solution for mean-payoff Streett games via their polynomial reduction to energy games [20].

#### 1.2 Contributions

Based on the above discussion, we distinguish between 1-fair and 2-fair energy (resp. mean-payoff) games, emphasizing which player is additionally constrained by the strong transition fairness condition. We provide a unified framework to solve both classes of games via a novel, gadget-based approach. Our gadgets enable the translation of fair quantitative games into regular quantitative games on a linearly larger state space. This allows solving these games in pseudo-polynomial time, significantly improving the computational complexity for 1-fair quantitative games, compared to the naive approaches outlined above. The origins of these gadget-based techniques can be traced back to the works in [24] and [19], where similar gadgets were employed to convert stochastic parity and Rabin games with almost-sure winning conditions into regular parity and Rabin games. In [34], a gadget-based technique was developed that reduces fair parity games to (linearly larger) regular parity games, demonstrating a quasi-polynomial run time for these games. Our work extends this approach to fair games with quantitative objectives. Concretely, our contributions are as follows.

#### Fair Mean-Payoff Games:

- $\triangleright$  Complexity. Using gadgets, we reduce (1- and 2-) fair mean-payoff games to regular mean-payoff games on a game arena with linearly larger number of nodes and edges, and a maximum absolute weight of  $\mathcal{O}(n^2W)$ , where n is the number of nodes and W is the maximum absolute weight in the fair game arena. Following the state-of-the-art algorithm of [28], we obtain a complexity of  $\mathcal{O}(n^4mW)$  for solving the fair mean-payoff games for optimal value problem, and a complexity of  $\mathcal{O}(n^3mW)$  for threshold problem.
- ▷ Determinacy. The above reduction establishes that fair mean-payoff games are determined irrespective of which player has fairness constraints.
- ▷ Strategies. We show that, in 1-fair mean-payoff games, player 1 in general needs infinite memory strategies to win, although finite memory suffices for achieving suboptimal values. Memoryless strategies suffice for player 2. In 2-fair mean-payoff games, player 2 has finite memory winning strategies, whereas memoryless strategies suffice for player 1.

### Fair Energy Games:

- Determinacy. In contrast to fair mean-payoff games, we show that fair energy games (with unknown initial credit) are not determined in general. In particular, 2-fair energy games are not determined, whereas 1-fair energy games are. We argue that the lack of determinacy prevents us from constructing similar gadgets to reduce 2-fair energy games to regular energy games.
- $\triangleright$  Complexity. We introduce a gadget that reduces 1-fair energy games to regular energy games on an arena with linearly larger number of nodes and edges, and a maximum absolute weight of  $\mathcal{O}(n^3W)$ . Using the state-of-the-art algorithm of [31], we obtain a complexity of  $\mathcal{O}(n^4mW)$  for solving these games. For 2-fair energy games, we provide simple algorithms to compute the winning regions separately for both players. We show that in 2-fair energy games, player 1's winning region is the same as in the corresponding energy game where fair

- edges are treated as regular edges. Meanwhile, player 2's winning region is the same as in the corresponding 2-fair mean-payoff games with threshold 0. These reductions yield an  $\mathcal{O}(n^3mW)$  algorithm for solving 2-fair energy games.
- ▷ Strategies. We show that in fair energy games, the player restricted by transition fairness has finite memory winning strategies whereas memoryless strategies suffice for the respective other player.

In summary, our results show that strong transition fairness can be seamlessly incorporated into quantitative games without significant computational overhead. In fact, just as in the qualitative setting, this simple form of fairness comes *virtually for free* in the quantitative setting. Our key conceptual insight is that both the simplicity of strong transition fairness, and the effectiveness of gadget-based approaches extend beyond the  $\omega$ -regular domain.

### 2 Preliminaries

In this section, we introduce the basic notations used throughout the paper. We write  $\mathbb{Q}$  and  $\mathbb{N}$  to denote the set of rational numbers and the set of natural numbers including 0, respectively.

## 2.1 Weighted Game Arena

A two-player weighted game arena is a tuple G = (Q, E, w), where  $Q = Q_1 \uplus Q_2$  is a finite set of nodes,  $E \subseteq Q \times Q$  is a set of edges, and  $w : E \to [-W, W]$  is a weight function that assigns an integer weight to each edge in G, where  $W \in \mathbb{N}$  is the maximum absolute weight appearing in G. The nodes are partitioned into two sets,  $Q_1$  and  $Q_2$ , where  $Q_i$  is the set of nodes controlled by Player i for  $i \in \{1,2\}$  and  $Q_1 \cap Q_2 = \emptyset$ . For a node q, we write qE to denote the set  $\{e \in E \mid e = (q,q') \text{ for } q' \in Q\}$  of all outgoing edges from q and E(q) to denote all successors of q. W.l.o.g., we assume that all nodes have out-degree at least one, i.e.,  $E(q) \neq \emptyset$  for all  $q \in Q$ . In rest of the paper, we use circles and squares in a figure to denote nodes controlled by player 1 and player 2, respectively.

**Plays.** A play  $\tau = q_0q_1\ldots\in Q^\omega$  on G is an infinite sequence of nodes starting from  $q_0$  such that,  $\forall i\geq 0, (q_i,q_{i+1})\in E$ . We use notations  $\tau[0;i]=q_0\ldots q_i$ , and  $\tau[i;j]=q_i\ldots q_j$  to denote the finite prefix and infix of the play  $\tau$ , respectively. A node q is said to appear infinitely often in  $\tau$ , i.e.,  $q\in \mathrm{Inf}(\tau)$  if  $\forall i,\exists j\geq i,$  such that  $q_j=q$ . We naturally extend the notion of appearing infinitely often in a play for the edges. We let  $\mathrm{plays}(G)$  denote the set of all plays on G, and let  $\mathrm{plays}(G,q)$  denote the set of all plays starting from node q.

**Strategies.** A strategy  $\sigma$  for player  $i \in \{1,2\}$  (or, a player i-strategy) is a function  $\sigma: Q^* \cdot Q_i \mapsto E$  where for all  $H \cdot q \in Q^* \cdot Q_i$ ,  $\sigma(H \cdot q) \in qE$ . Intuitively, from every node  $q \in Q_i$ , a strategy for player i assigns an outgoing edge of that node based on a history  $H \in Q^*$ .

For a player i strategy  $\sigma$ , a play  $\tau = q_0 q_1 \dots$  is called a  $\sigma$ -play if it conforms with  $\sigma$ , i.e., for all  $j \in \mathbb{N}$  with  $q_j \in Q_i$ , it holds that  $\sigma(q_0 \dots q_j) = (q_j, q_{j+1})$ . Given a strategy  $\sigma$  we denote the restriction of sets  $\mathsf{plays}(G)$  and  $\mathsf{plays}(G,q)$ 

to  $\sigma$ -plays with  $\operatorname{plays}_{\sigma}(G)$  and  $\operatorname{plays}_{\sigma}(G,q)$ . Similarly,  $\operatorname{play}_{\sigma,\pi}(G,q)$  denotes the unique play from q conforming with player 1 and player 2 strategies  $\sigma$  and  $\pi$ .

Let M be a set called memory. A player i strategy  $\sigma$  with memory M can be represented as a tuple  $(M, m_0, \alpha, \beta)$ , where  $m_0 \in M$  is the initial memory value,  $\alpha: M \times Q \to M$  is the update function, and  $\beta: M \times Q_i \to Q$  is the function prescribing next state. Intuitively, if the current node is a player i node q and m is the current memory value, the strategy  $\sigma$  selects  $q' = \beta(m,q)$  as the next node and updates the memory to  $\alpha(m,q)$ . If M is finite, then we call  $\sigma$  a finite memory strategy; otherwise it is an infinite memory strategy. Formally, given a history  $H \cdot q \in Q^* \cdot Q_i$ ,  $\sigma(H \cdot q) = \beta(\hat{\alpha}(m_0, H), q)$ , where  $\hat{\alpha}$  extends  $\alpha$  to sequences of nodes canonically. A strategy is called memoryless or positional if |M| = 1. For such memoryless strategy  $\sigma$ , it holds that  $\sigma(H_1 \cdot q) = \sigma(H_2 \cdot q)$  for every history  $H_1, H_2 \in Q^*$ . We denote it as  $\sigma(q)$  for convenience. For any positional strategy  $\sigma$  of player i, we define  $G_{\sigma}$  to be the subgame arena (Q, E', w) of G, where  $E' = \{(q, \sigma(q)) \mid q \in Q_i\} \cup \{qE \mid q \in Q_{3-i}\}$ . Note that in this subgame arena, each player i node has exactly one successor according to the strategy  $\sigma$ .

Weighted Games and Objectives. A weighted game is a tuple  $(G, \varphi)$ , where G is a weighted game arena and  $\varphi \subseteq Q^{\omega}$  is an objective for player 1. A play  $\tau$  is winning for player 1 if  $\tau \in \varphi$ , else it is winning for player 2. A player i strategy  $\sigma$  is winning from some node q, if all  $\sigma$ -plays starting from q are winning for player i. We denote the winning regions of player i by  $\text{Win}_i(G, \varphi)$ , or shortly  $\text{Win}_i$ , when  $(G, \varphi)$  is clear from the context. Formally, the winning regions are defined as follows where  $\Sigma_i$  is the set of all player i strategies:

$$q \in \operatorname{Win}_1(G, \varphi) \iff \exists \sigma \in \Sigma_1. \, \forall \pi \in \Sigma_2, \, \operatorname{play}_{\sigma, \pi}(G, q) \in \varphi$$
 (1)

$$q \in \operatorname{Win}_2(G, \varphi) \iff \exists \pi \in \Sigma_2. \, \forall \sigma \in \Sigma_1, \, \operatorname{play}_{\sigma, \pi}(G, q) \notin \varphi$$
 (2)

A game  $(G, \varphi)$  is called determined, if  $Q = Win_1 \cup Win_2$ .

Given a finite infix  $H=q_0\dots q_k$  of a play in G, we denote the (total) weight and the average weight of H by  $w(H)=\sum_{i=0}^{k-1}w(q_i,q_{i+1})$  and  $\operatorname{avg}(H)=\frac{w(H)}{k}$ , respectively. Furthermore, we denote the (limit) average weight of a play  $\tau$  by  $\operatorname{avg}(\tau)=\liminf_{i\to\infty}\operatorname{avg}(\tau[0;i])$ .

**Mean-Payoff Games.** A mean-payoff game is a weighted game with a mean-payoff objective for a given threshold value  $v \in \mathbb{Q}$ , which is defined as  $MP_v = \{\tau \in Q^\omega \mid \operatorname{avg}(\tau) \geq v\}$ . Intuitively, a play is winning for player 1 if the average weight of the play is above a certain threshold value v. Note that, mean-payoff games are  $\operatorname{prefix}$  independent as any finite prefix of a play  $\tau$  does not affect the limit average weight of  $\tau$ .

In this work, without loss of generality, we focus on mean-payoff objective  $MP_0$  with threshold value v=0. It is easy to see that for any value v, the weighted mean-payoff game  $(G, MP_v)$  with G=(Q, E, w) can be reduced to a mean-payoff game  $(G', MP_0)$  by subtracting v from the weights of all edges in G, i.e., G'=(Q, E, w') where w'(q, q')=w(q, q')-v for all  $(q, q') \in E$ .

The *optimal value* of a mean-payoff game is the maximum value v for which player 1 has a winning strategy with threshold v. A winning strategy of player 1

that achieves this optimal value is called an *optimal value strategy*. The *optimal value problem* is to compute the optimal values in a mean-payoff game.

Energy Games. An energy objective w.r.t. a given initial credit  $c \in \mathbb{N}$  is defined as  $\operatorname{En}_c = \{\tau \in Q^\omega \mid c + w(\tau[0;i]) \geq 0, \ \forall i \in \mathbb{N}\}$ . Intuitively, a play belongs  $\operatorname{En}_c$  if the total weight ('energy level' starting from c) of the play remains non-negative along the play. An energy game is a weighted game with an energy objective with unknown initial credit, denoted by  $\operatorname{En}_c$ , where player 1 wins from some node q if there exists an initial credit  $c \in \mathbb{N}$  such that she can ensure the objective  $\operatorname{En}_c$  from q. Formally, the winning regions are defined as follows, for  $\varphi_c = \operatorname{En}_c$ :

$$q \in \mathrm{Win}_1 \iff \exists c \in \mathbb{N}. \, \exists \sigma \in \varSigma_1. \, \forall \pi \in \varSigma_2, \, \, \mathrm{play}_{\sigma,\pi}(G,q) \in \varphi_c \qquad \qquad (3)$$

$$q \in \text{Win}_2 \iff \exists \pi \in \Sigma_2. \, \forall c \in \mathbb{N}. \, \forall \sigma \in \Sigma_1, \, \text{play}_{\sigma,\pi}(G,q) \notin \varphi_c$$
 (4)

Note that, in contrast to mean-payoff games, the energy objective is not a prefix-independent objective as the total weight of each prefix has to be nonnegative along any play. However, it is known that such energy games are log-space equivalent to mean-payoff games [15]. Furthermore, it is known that both energy games and mean-payoff games are positionally determined [17,32] i.e.,  $Q = \mathtt{Win}_1 \cup \mathtt{Win}_2$  and both players have positional winning strategies.

#### 2.2 Fair Game Arena

A fair game arena is a tuple  $(G, E_f)$ , where G is a weighted game arena as defined above and  $E_f \subseteq E$  is a given set of 'fair edges'. We call the non-fair edges  $E \setminus E_f$  'regular', and a node q 'fair' if it is the source node of a fair edge, i.e.,  $qE \cap E_f \neq \emptyset$ . Let  $Q_f$  be the set of all fair nodes in G. We investigate the scenario where all fair nodes are owned by the same player. If  $Q_f \subseteq Q_1$ , we call the game arena 1-fair and if  $Q_f \subseteq Q_2$ , we call it 2-fair. For brevity, we sometimes denote the fair game arena  $(G, E_f)$  by G and keep the set of fair edges  $E_f$  implicit. Given a fair node q, we write  $qE_f$  to denote the set  $\{e \in E_f \mid e = (q, q') \text{ for } q' \in Q\}$  of all fair outgoing edges from q and  $E_f(q) = \{q' \mid (q, q') \in E_f\}$  to denote the set of successors of q via fair edges.

A play  $\tau$  is fair if for all nodes  $q \in \mathtt{Inf}(\tau)$  and for all edges  $e \in qE$ , if  $e \in E_f$ , then  $e \in \mathtt{Inf}(\tau)$ . We let  $\mathtt{plays}^f(G)$  denote the set of all fair plays on  $(G, E_f)$ . We say a strategy  $\sigma$  for player i is fair if  $\mathtt{plays}_{\sigma}(G) \subseteq \mathtt{plays}^f(G)$ .

Fair Mean-Payoff/Energy Games. An *i-fair mean-payoff game* with objective  $\varphi = MP_v$  is a tuple  $(G, E_f, \varphi^f)$ , where  $(G, E_f)$  is an *i-fair* game arena, and  $\varphi^f$  is the objective with the respective fairness condition defined as follows:

$$\varphi^f = \begin{cases} \varphi \cap \mathsf{plays}^f(G) & \text{if } (G, E_f) \text{ is 1-fair} \\ \varphi \cup \mathsf{plays}(G) \setminus \mathsf{plays}^f(G) & \text{if } (G, E_f) \text{ is 2-fair} \end{cases}$$
(5)

Intuitively, a play is winning for player i in i-fair game if it satisfies both the fairness condition and the player's corresponding objective, i.e.,  $\varphi$  for player 1 and  $\neg \varphi$  for player 2.

Similarly, an *i-fair energy game* with energy objective En is a tuple  $(G, E_f, \operatorname{En}^f)$  where the winning regions are defined as in (3)-(4) for  $\varphi_c = \operatorname{En}_c^f$ .

Anand et al.

# 3 Example: Mean-Payoff vs Fair Mean-Payoff Games

In this section, we depict the intricacy of fair mean-payoff games in comparison to regular mean-payoff games via the following example:

Consider the game are na depicted in Fig. 1 with both the nodes belonging to player 1. With regular mean-payoff objective with threshold 0, player 1 has a simple optimal positional strategy: at q, play the self-loop (q,q) and from p play the edge (p,q). This strategy is winning for player 1 because the limit average weight of any play conforming to this strategy is 1. Note that the strategy is also an optimal value strategy as the optimal value of the game is 1.

Now let us consider the game arena to be 1-fair with a fair edge (q, p). Now the fair mean-payoff objective  $MP_0^f$  for player 1 is to ensure that the limit average weight of the play is  $\geq 0$  along with the fairness condition that says if q appears infinitely often in a play, so does the edge (q, p).



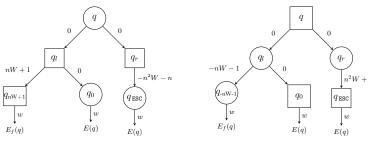
Fig. 1

First note that there is no positional strategy that can ensure both fairness and mean-payoff objective and hence winning for player 1. Now, consider the following finite memory strategy for player 1: the strategy takes the self-loop (q,q) for k times before taking (q,p) once, and this sequence of k+1 choices is repeated forever. The resulting play uses the fair edge (q,p) infinitely often and hence it is fair. For  $k \geq 4$ , this strategy is winning as the limit average of the plays conforming to this strategy is at least 0. In fact, we can show that for all  $\epsilon > 0$ , it is possible to choose a large enough k, such that player 1 ensures a limit average weight of at least  $1 - \epsilon$ .

Finally, consider the following infinite-memory strategy for player 1: the strategy is played in rounds; in round  $i \geq 0$ , the strategy plays (q,q) for i times, then plays (q,p) once and progresses to round i+1. This fair strategy ensures player 1 achieves a limit average weight of value 1, and it is therefore an optimal value strategy. However, it is not hard to see that there is no finite-memory optimal value strategy for player 1 in this game. In the following sections, we formally present how to solve fair mean-payoff games and discuss the memory requirements for each player to achieve these winning strategies.

## 4 Solving Fair Mean-Payoff Games

In this section, we present algorithms for solving fair mean-payoff games. The algorithms transform a fair mean-payoff game into a 'regular' mean-payoff game using gadgets. We introduce two gadgets depending on which player owns the fair edges as depicted in Fig. 2. Given an i-fair mean-payoff game G, we replace all fair nodes with the corresponding i-fair gadgets and obtain an equivalent mean-payoff game G' such that  $(G, E_f, MP_0^f)$  is winning for player i if the mean-payoff game  $(G', MP_0)$  is winning for player i. In particular, we prove the following.



(a) 1-fair mean-payoff gadget

(b) 2-fair mean-payoff gadget

Fig. 2: Gadgets for converting fair mean-payoff games to mean-payoff games. Edges are labeled with their weights. The edges labeled with w represent: an edge from a gadget node to a node  $q' \in E(q)$  or  $E_f(q)$  carry the weight w(q, q').

**Theorem 1.** Let  $(G, E_f, MP_0^f)$  be a fair mean-payoff game with threshold 0, where G = (Q, E, w),  $w : E \to [-W, W]$ , and |Q| = n. Then there exists a mean-payoff game G' = (Q', E', w') where  $Q' \supseteq Q$ ,  $|Q'| \le 6n$  and  $w' : E' \to [-W', W']$  with  $W' = n^2W + n$  such that  $Win_i(G) = Win_i(G') \cap Q$  for  $i \in \{1, 2\}$ .

Standard algorithms for mean-payoff games [28] with Thm. 1 gives an  $\mathcal{O}(n^3 mW)$  time algorithm (with |E| = m) for solving fair mean-payoff games with threshold and an  $\mathcal{O}(n^4 mW)$  time algorithm for the optimal value problem. As regular mean-payoff games are determined, we also get the following from Thm. 1:

**Theorem 2.** Fair mean-payoff games are determined.

The rest of this section discusses the proof of Thm. 1 for 1-fair mean-payoff games in Sec. 4.1 and for 2-fair mean-payoff games in Sec. 4.2. The detailed proof of Thm. 1 for 2-fair mean-payoff games is provided in App. B. Finally, Sec. 4.3 discusses the memory requirements of strategies in fair mean-payoff games resulting from the gadget-based reductions.

## 4.1 Proof of Thm. 1 for 1-fair Mean-Payoff Games

**Gadget Construction & Intuition.** Let  $(G, E_f, MP_0^f)$  be a 1-fair mean-payoff game. We construct the *gadget game* G' by replacing every fair node in  $q \in Q_f$  in G, with its 3-step gadget presented in Fig. 2a. That is, the incoming edges of q is redirected to node q at the root, and the outgoing edges on the third step lead to  $E_f(q)$  and E(q), the fair successors and successors of q, respectively. The formal construction of G' can be found in App. A.

For notational convenience, for each  $q \in Q_f$ , we denote the leftmost branch  $q \to q_l \to q_{nW+1}$  of the gadget as fair branch  $\operatorname{br}_{\mathtt{fair}}^q$ , the middle branch  $q \to q_l \to q_0$  as the simulation branch  $\operatorname{br}_{\mathtt{sim}}^q$  and the rightmost branch  $q \to q_r \to q_{\mathtt{ESC}}$  as the escape branch  $\operatorname{br}_{\mathtt{esc}}^q$ . Intuitively, from a fair node q, player 1 can escape to a part of the game that doesn't contain q and is player 1 winning (according to the

current player 1 strategy) by taking the escape branch and therefore paying a high negative payoff. Since by this escape choice q is guaranteed to be seen only finitely often, the negative payoff associated with the escape edge does not change the winner of the game. Now assume there is no such winning escape choice for player 1 from q. Then from q, all player 1 winning play visits q infinitely often, and thus is forced to visit all of q's outgoing fair edges infinitely often. If taking one of its fair outgoing edges pushes the game into a player 2 winning region that does not contain q (according to some player 2 strategy), then in the gadget game player 2 can pay a high positive payoff and choose the fair branch with the correct successor to escape to a player 2 winning region. So, in a way "fair branch is the escape branch for player 2". However, player 2 can take this escape branch to win from q, only if player 1 cannot take her escape branch to win. The difference in the amplitude of weights between the escape branches of the two players (i.e.  $br_{esc}^q$  and  $br_{fair}^q$ ) stems from this hierarchy between the escape choices of the players. If neither player has an escape choice, then the middle branch  $\mathtt{br}^q_{\mathtt{sim}}$  faithfully simulates the fair mean-payoff game without adding any additional weights to the play via the gadget edges to alter the winner.

Using this intuition, we now prove both directions of Thm. 1 separately for 1-fair mean-payoff games.

**1. Proof of**  $(Win_1(G') \cap Q \subseteq Win_1(G))$ : Let  $\sigma'$  be a positional player 1 strategy that wins q in G' for the regular mean-payoff objective. We construct a player 1 strategy  $\sigma$  that wins q in G.

Consider the subgame arena  $G'_{\sigma'}$  of G'. All cycles in  $G'_{\sigma'}$  are non-negative; else player 2 can force to eventually only visit a negative cycle and therefore construct a play in  $G'_{\sigma'}$  with negative mean-payoff value.

Using  $\sigma'$ , we construct  $\sigma$  in G as follows:

- a. If  $q' \notin Q_f$ , then set  $\sigma$  to be positional at q' with  $\sigma(q') = \sigma'(q')$ .
- b. If  $\sigma'(q') = q_r$  for some fair node  $q' \in Q_f$ , then set  $\sigma$  to be positional at q' with  $\sigma(q') = \sigma'(q'_{ESC})$ . Intuitively this says if for a fair node q',  $\sigma'$  asks to choose the escape branch of the gadget, then  $\sigma$  follows  $\sigma'$  to choose that branch and moves to the corresponding 'escape node' in G.
- c. If  $\sigma'(q') = q_l$  for some fair node  $q' \in Q_f$ , then we set  $\sigma$  to be an infinite memory strategy at q' that is played in rounds: consider an order on the fair successors of q'. In round  $i \geq 0$ ,  $\sigma$  chooses the 'preferred successor'  $\sigma'(q'_0)$  exactly i times and then chooses the i<sup>th</sup> (modulo  $|E_f(q')|$ ) fair successor.

Now we prove that all  $\tau \in \mathtt{plays}_{\sigma}(G)$  are winning for player 1 in G. In particular, we will show that  $\tau$  is fair and  $\mathtt{avg}(\tau) \geq 0$ .

For a play  $\tau=q^0q^1\ldots\in \mathtt{plays}_\sigma(G)$ , we construct its extension  $\tau'$  in G' inductively as follows: we start with  $q^0$  and i=0 and with increasing i,

- If  $q^i \notin Q_f$ , append  $\tau'$  with  $q^{i+1}$ ;
- If  $\sigma(H \cdot q^i)$  is defined using Item b. i.e.,  $\sigma'(q^i) = q_r^i$  then append  $\tau'$  with  $\operatorname{br}_{\tt esc}^{q^i} \to q^{i+1}$ . In particular, we append  $\tau'$  with  $q_r^i \cdot q_{\tt ESC}^i \cdot q^{i+1}$ .
- If  $\sigma(H \cdot q^i)$  is defined using Item c., then

- $$\begin{split} \bullet & \text{ if } q^{i+1} = \sigma'(q^i_0) \text{, append } \tau' \text{ with } \mathsf{br}^{q^i}_{\mathtt{sim}} \to q^{i+1} \text{ i.e., with } q^i_l \cdot q^i_0 \cdot q^{i+1} \text{ ;} \\ \bullet & \text{ else, append } \tau' \text{ with } \mathsf{br}^{q^i}_{\mathtt{fair}} \to q^{i+1} \text{, i.e. } q^i_l \cdot q^i_{\mathtt{nW}+1} \cdot q^{i+1}. \end{split}$$

Clearly,  $\tau'$  is a play in  $G'_{\sigma'}$ ; hence it contains only non-negative cycles. Further, the restriction of  $\tau'$  to the nodes Q of G, denoted by  $\tau'|_Q$ , is exactly  $\tau$ .

Let a 'tail' of  $\tau$  denote an infinite suffix  $\hat{\tau}$  of  $\tau$  that contains only the nodes and edges visited infinitely often in  $\tau$ ; and let  $\hat{\tau}'$  be the extension of  $\tau$  in G'. We denote the subgraph of  $G'_{\sigma'}$  restricted to the nodes and edges in  $\hat{\tau}'$  by  $G'_{\hat{\tau}'}$ .

We first show that  $\tau$  is fair. It is sufficient to show that  $\hat{\tau}$  does not contain fair nodes p for which  $\sigma(p)$  is defined via Item b., because for all the other fair nodes in  $\hat{\tau}$ ,  $\tau$  visits all their fair successors infinitely often (Item c.). Let p be a node for which  $\sigma(p)$  is defined via Item b.. Then  $\sigma'(p) = p_r$ . As  $\sigma'$  is positional, only the escape branch of p exists in  $G'_{\sigma'}$ , carrying the weight  $-n^2W-n$ . Since p is visited infinitely often, there is a (simple) cycle in  $G'_{\sigma'}$  that passes through p, and thus sees the weight  $-n^2W-n$ . In order to maintain a non-negative weight, the cycle needs to contain n-many fair branches (carrying weight nW+1), which is not possible since there are at most n-1 many fair branches in  $G'_{\sigma'}$ . With this we conclude that  $\tau$  is fair.

Now we prove that  $avg(\tau) \geq 0$ . First let us make an observation about the cycles in  $G'_{\sigma'}$  that use only simulation branches.

**Observation** (sim-cycles) We call cycles  $\mathbf{c}'$  in  $G'_{\sigma'}$  that do not contain any fair or escape branches sim-cycles. That is,  $\mathbf{c}'$  contains only simulation branches. Being part of  $G'_{\sigma'}$ , **c** has non-negative weight. Since all the gadget branches in  $\mathbf{c}'$  carry 0 weight, the restriction  $\mathbf{c} = \mathbf{c}'|_Q$  is a non-negative weight cycle in G.

We saw that  $\hat{\tau}$  does not contain nodes defined using Item b.. I.e., for all fair nodes p in  $\hat{\tau}$ ,  $\sigma$  is defined via Item c., and  $G'_{\hat{\tau}'}$  contains the branches  $\operatorname{br}_{\mathtt{fair}}^p$ and/or  $br_{sim}^p$ . Thus, all cycles in  $\hat{\tau}'$  are either sim-cycles, or visit a fair branch.

If  $\hat{\tau}$  does not contain any fair nodes,  $\hat{\tau} = \hat{\tau}'$ , and  $avg(\tau) \geq 0$  easily follows. Now we assume  $\hat{\tau}$  contains at least one fair node, and use the following easy-toverify remark to show  $avg(\tau) > 0$ :

Remark 1. Given a play  $\rho$  and  $\epsilon > 0$ , if there exists a  $k \in \mathbb{N}$  and a tail  $\hat{\rho}$  of  $\rho$ such that every k-length infix  $\hat{\rho}_k$  of  $\hat{\rho}$  satisfies  $avg(\hat{\rho}_k) \geq -\epsilon$ , then  $avg(\hat{\rho}) \geq -\epsilon$ .

Let  $\epsilon > 0$  be arbitrary. Let  $k \in \mathbb{N}$  such that  $\frac{n^2W}{k} < \epsilon$ . Since  $\hat{\tau}$  has at least one fair node, Item c. is triggered infinitely often to construct the strategy  $\sigma$ . Therefore, the round (in the sense of Item c.) will grow unboundedly bigger. Thus, for any k, we can get a tail  $\hat{\tau}$  of  $\tau$  that is in round k. In  $\hat{\tau}$ , for every fair node, their edge to their preferred successor is taken at least k-many times between any two occurrences of their other (fair) outgoing edges. Then in  $\hat{\tau}'$ , the simulation branches of fair nodes are visited at least k-many times in between two occurrences of fair branches. Therefore, in  $\hat{\tau}'$  each infix  $\hat{\tau}'_k$  of length k visits the fair branch of each node at most once. Consequently, there are at most n-many simple cycles in  $\hat{\tau}'_k$  that visits a fair branch, and all its other simple cycles are sim-cycles. Now, total weight of n-many simple cycles that are not

sim-cycles are  $\geq -n^2W$ , and since all the sim-cycles have non-negative weight,  $\operatorname{avg}(\tau_k) \geq \frac{-n^2 \overline{W}}{k} \geq -\epsilon$ . Using the fact from the previous paragraph, we get that  $\operatorname{avg}(\tau) \geq -\epsilon$ . This implies that for any  $\epsilon > 0$ ,  $\operatorname{avg}(\tau) \geq -\epsilon$ . Hence,  $\operatorname{avg}(\tau) \geq 0$ . This concludes the proof of the first direction.

**2.** Proof of  $(Win_2(G') \cap Q \subseteq Win_2(G))$ : Analogous to the previous part, we consider a positional winning strategy  $\pi'$  of player 2 in G' and construct, this time, a positional winning strategy  $\pi$  for player 2 in G. Note that, none of the fair nodes in G belong to player 2 and hence the construction of  $\pi$  is quite straight forward: for all  $q \in Q_2$ , we set  $\pi(q) = \pi'(q)$ . We define  $G'_{\pi'}$  as the subgame restricted to  $\pi'$ , as before.

For a play  $\tau = q^0 q^1 \dots \in \mathtt{plays}_\pi(G)$ , we define its extension  $\tau'$  in G' inductively as follows: we start from with  $q^0$  and i=0, and with increasing i,

- If  $q^i \notin Q_f$ , append  $\tau'$  with  $q^{i+1}$ ; Otherwise,
- If  $\operatorname{br}_{\mathtt{sim}}^{q^i}$  exists in  $G'_{\pi'}$ , append  $\tau'$  with  $q_l^i \cdot q_0^i \cdot q^{i+1}$ ;
- $\begin{array}{l} \text{ If } \mathtt{br}^{q^i}_{\mathtt{sim}} \text{ does not exist in } G'_{\pi'} \colon \\ \bullet \text{ If } q^{i+1} = \pi'(q^i_{\mathtt{nW}+1}), \text{ append } \tau' \text{ with the fair branch i.e. } q^i_l \cdot q^i_{\mathtt{nW}+1} \cdot q^{i+1}, \\ \bullet \text{ Otherwise, append it with the escape branch, i.e. } q^i_r \cdot q^i_{\mathtt{ESC}} \cdot q^{i+1}. \end{array}$

We define the tail of  $\tau$  as  $\hat{\tau}$  and its extension  $\hat{\tau}'$ , analogous to the previous part. We denote the subgraph of  $G'_{\tau'}$  restricted to the nodes and edges in  $\hat{\tau}'$  by  $G'_{\hat{\tau}'}$ .

Let  $\tau \in \mathsf{plays}_{\pi}(G)$  be a play in G that conforms with  $\pi$ . If  $\tau$  is not fair, it is automatically won by player 2. Therefore, assume that  $\tau$  is fair. We show that all the simple cycles in  $\hat{\tau}$  has negative weight. For this, we use the fact that its extension  $\tau'$  is a play in  $G'_{\pi'}$  and hence all the cycles in  $G'_{\hat{\tau}'}$  have negative weight. Since the length of simple cycles are bounded by n, this gives us  $avg(\tau) < 0$ . The main argument is the absence of fair branches in  $G'_{\hat{\tau}'}$ , introduced in Claim 1.

# Claim 1 If $\tau$ is fair, there exist no fair branches $(\operatorname{br}_{fair})$ in $G'_{\hat{\tau}'}$ .

Before proving the claim, we discuss how it implies that all simple cycles in  $\hat{\tau}$  are negative, resulting  $avg(\tau) < 0$ . From the definition of extension, absence of fair branches in  $G'_{\hat{\tau}'}$  implies that either (i) for all  $q \in Q_f$  only  $\operatorname{br}_{\operatorname{sim}}^q$  exists in  $G'_{\hat{\tau}'}$ , or (ii) for some  $q \in Q_f$ , only  $\operatorname{br}_{\operatorname{esc}}^q$  exists in  $\hat{\tau}'$ . By construction of  $\tau'$ , case (ii) above implies the existence of a fair node  $q \in Q_f$  that appears infinitely often in  $\tau$  but does not take all its fair outgoing edges, in particular  $q \to \pi'(q_{nW+1}) \in E_f(q)$ infinitely often. This makes  $\tau$  unfair, contradicting our assumption. In case of (i), since all simple cycles  $\mathbf{c}'$  in  $\hat{\tau}'$  are negative and none of the gadget nodes contribute any weight due to only  $\mathtt{br_{sim}}$  existing in  $\hat{\tau}'$ , we conclude that the cycles  $\mathbf{c} = \mathbf{c}'|_Q$  in  $\hat{\tau}$  have negative weight. This implies that  $\mathtt{avg}(\tau) < 0$ .

By the above-mentioned argument, proving Claim 1 concludes the second direction of the proof.

Proof of Claim 1. As discussed above, whenever  $\operatorname{br}_{\operatorname{esc}}^q$  is in  $G'_{\hat{\tau}'}$  for a node q,  $\mathtt{br}^q_{\mathtt{fair}}$  is also in it. Now we remove all the escape branches from  $G'_{\hat{\tau}'}$  and obtain the subgraph S. Note that S has no dead-ends, since for all q for which  $br_{esc}^q$ got removed from  $G'_{\hat{\tau}'}$ ,  $\operatorname{br}_{\operatorname{fair}}^q$  was also in  $G'_{\hat{\tau}'}$ .

For all nodes q in S, either  $\operatorname{br}_{\mathtt{fair}}^q$  or  $\operatorname{br}_{\mathtt{sim}}^q$  is in S. As a subgraph of  $G'_{\pi'}$ , all cycles in S have negative weight. It is easy to see that a fair branch cannot lie on a simple cycle in S, because fair branches carry weight nW+1 and the other edges come either from  $\operatorname{br}_{\mathtt{sim}}$  carrying weight 0, or from G carrying weight  $\geq -W$ . Hence, a simple cycle containing a fair branch would have positive weight.

Now assume fair branches exist in  $\hat{\tau}'$ , but not on cycles of S. Then all cycles in S consist only of simulation branches. However, nodes with simulation branches have the same outgoing edges in  $G'_{\hat{\tau}'}$  and in S. Since every node in S is visited infinitely often in  $\tau$ ,  $\tau$  will eventually enter cycles of S and never leave them, since these nodes have the same outgoing edges in  $G'_{\hat{\tau}'}$ . This implies fair branches are not infinitely often visited in  $\tau$ , and concludes the proof of Claim 1.

### 4.2 Proof of Thm. 1 for 2-fair Mean-Payoff Games

Observing page constrains, we provide the proof of Thm. 1 for 2-fair mean-payoff games in App. B and only discuss its differences to Sec. 4.1 here.

Interestingly, the player 2 gadget for 2-fair games (Fig. 2b) is exactly the dual of the player 1 gadget for 1-fair games (Fig. 2a). This is surprising, as the winning objectives of player 1 and 2 are not exactly dual in these games. While player 1 in 1-fair games is expected to win the vertices with optimal value 0 (that is, the best value a player 1 strategy can achieve from these nodes is 0, and getting  $\epsilon$ -close to the best possible value isn't sufficient to win); for player 2 to win a vertex in a 2-fair game, it is sufficient for him to get  $\epsilon$ -close to the best possible value he can achieve from this vertex. As we will see in the next chapter, these differences in the objectives' behavior w.r.t. optimal values reflect drastically in the required strategy sizes. Namely, in 2-fair games finite strategies are sufficient whereas in 1-fair games infinite strategies are required to win.

It is therefore surprising that this imbalance in objectives does not demand any changes in the gadget's structure for proving Thm. 1 in the case of 2-fair mean-payoff games. The main difference w.r.t. the proof form Sec. 4.1 lies in the construction of winning strategies for the fair player. In particular, the proof for 1-fair games (in Sec. 4.1) is slightly more complicated due to the required infinite strategy construction for player 2. On the other hand, the proof of 2-fair games (in App. B) reveals that a finite memory is sufficient for a winning player 2 strategy in 2-fair games.

#### 4.3 Strategy Complexity for Fair Mean-Payoff Games

We list an overview of results on strategy requirements for fair mean-payoff games. Most of these results follow from the proofs of Thm. 1. We discussed in Sec. 3 that player 1 may need infinite memory to achieve an optimal value in a 1-fair mean-payoff game. Lem. 1 shows that player 1 can reach  $\epsilon$ -close to the optimal value with finite memory strategies.

**Lemma 1.** Given a 1-fair mean-payoff game  $(G, E_f, MP^f)$ , let the optimal value from some node  $q_0$  is v. Then, for all  $\epsilon > 0$ , there exists a finite-memory strategy of player 1 that is winning from  $q_0$  in fair mean-payoff game  $(G, E_f, MP(v - \epsilon))$ .

*Proof.* W.l.o.g, we show this for v=0. In the proof of Thm. 1, we construct an infinite memory strategy  $\sigma$  for player 1 where at  $i^{\text{th}}$  round, player 1 plays the preferred successor i times and then plays a fair successor. Any  $\sigma$  play has the following property: for any  $\epsilon$ , there exists a tail of the play such that the average weight of every k-length infix of the tail is at least  $-\epsilon$ . We fix this k and modify the strategy construction in Item c. to play the preferred successor k times and then play a fair successor in every round. By Rem. 1, this finite memory strategy ensures the mean-payoff value of any play is at least  $-\epsilon$ .

The above lemma entails that in a 1-fair mean-payoff game with threshold value 0, player 1 has a finite memory strategy from a node  $q \in \text{Win}_1$  if the optimal value is strictly larger than 0. On the other hand, in 2-fair mean-payoff games getting  $\epsilon$ -close to the best value achievable from a vertex is always sufficient to ensure winning for player 2. Intuitively, for this reason, player 2 has finite memory winning strategies in 2-fair mean-payoff games.

**Lemma 2.** Given a 2-fair mean-payoff game  $(G, E_f, MP^f)$ , for all  $q \in Win_2$ , there exists a finite-memory strategy of player 2 that is winning from q.

Finally, the proofs of Thm. 1 from Sec. 4.1 (resp. App. B) reveal the existence of memoryless winning player 2 (resp. player 1) strategies in 1-fair (resp. 2-fair) mean-payoff games.

**Lemma 3.** For all fair mean-payoff games, memoryless strategies are sufficient for the player who does not own the fair nodes.

### 5 Solving Fair Energy Games

In this section, we aim to extend our gadget-based approach to solve fair energy games. Intuitively, gadget-based approaches transfer the determinacy of regular (qualitative or quantitative) objectives to their fair variants, as demonstrated in [34] for fair parity games and by the proofs of Thm. 1 for fair mean-payoff games. Conceptually, these techniques build on finding winning strategies for either player in the fair game – based on their winning strategies in the gadget game – with similar infinite behavior. Therefore, before attempting a gadget-based approach for fair games, we need to ensure that these games are determined.

In this section, we show that 2-fair energy games are *not determined*, whereas 1-fair energy games are. While due to the above-mentioned reason we cannot construct gadgets for 2-fair energy games, we give simple algorithms to compute both  $Win_1$  and  $Win_2$ . For 1-fair energy games, we observe that the gadget in Fig. 2b falls short, and we present new gadgets.

#### 5.1 Discussion on Determinacy of Energy Games

Whenever  $\varphi_c$  is a Borel set, the following holds due to Borel determinacy [38]:

$$\forall \sigma \in \Sigma_1. \, \exists \pi \in \Sigma_2, \, \mathtt{play}_{\sigma,\pi}(G,q) \not \in \varphi_c \Leftrightarrow \exists \pi \in \Sigma_2. \, \forall \sigma \in \Sigma_1, \, \mathtt{play}_{\sigma,\pi}(G,q) \not \in \varphi_c$$

As  $\operatorname{En}_c$  and  $\operatorname{En}_c^f$  are Borel sets, the equation holds, showing that energy and fair energy games are determined under a fixed credit c. Furthermore, this equality combined with the negation of Eq. (1) defining  $\operatorname{Win}_1$ , yields the following formulation of  $Q \setminus \operatorname{Win}_1$ :

$$q \notin \text{Win}_1 \iff \forall c \in \mathbb{N}. \, \exists \pi \in \Sigma_2. \, \forall \sigma \in \Sigma_1, \, \text{play}_{\sigma,\pi}(G,q) \notin \varphi_c$$
 (6)

Again, this formulation holds for both energy and fair energy games. Clearly it follows that a (fair) energy game is determined if and only if Eq. (4) is equivalent to Eq. (6). In fact, if we can restrict the quantification over c in Eq. (6) to a finite set, then we can swap  $\forall c \in \mathbb{N}$  and  $\exists \pi \in \Sigma_2$ , which yields the desired equivalence, showing the game is determined. This is indeed the case in energy games: Whenever player 1 wins from a node, it wins with initial credit nW. Dually, whenever player 2 wins w.r.t. initial credit nW, it wins against every initial credit. That is, the quantification over the c can be restricted to [0, nW].

Using this trick, we show in Sec. 5.3 that 1-fair energy games are determined. However, we demonstrate in the next section that this does not hold for 2-fair energy games and that these games are not determined.

#### 5.2 2-fair Energy Games

We start by showing that 2-fair energy games are not determined.

**Theorem 3.** 2-fair energy games are not determined.

Proof. Recall that energy games are determined if and only if Eq. (4) is equivalent to Eq. (6). We provide a counterexample to this equivalence by the game graph in Fig. 3. Any fair strategy  $\pi$  for player 2 takes the edge (q,q') after finitely many steps, say after c steps. Then, the unique strategy  $\sigma$  for player 1, which takes the self-loop on q', wins the play  $\operatorname{play}_{\sigma}(G,q)$  with respect to the initial credit c. Thus, player 2 has no winning strategy. However, player 1 has no winning strategy either since for any initial credit c, there exists a player 2 strategy that ensures the energy of every play drops below 0. Namely, any strategy that takes the self-loop of q more than c times achieves this.

This occurs as player 2 can delay taking his fair edges enough to violate any given c while still satisfying fairness condition in the suffix. Hence, if player 2 can force a negative cycle, then he can use it to violate any initial credit. In contrast, player 1's objective remains the same as in the regular energy game: she



Fig. 3

wins from a node iff she can prevent player 2 from forcing a negative cycle from that node. This leads to the following result, which is proven in App. C.

**Theorem 4.** Given a 2-fair energy game  $(G, E_f, E_n)$ , the set of winning nodes for player 1 is the same as the set of winning nodes for player 1 in the (regular) energy game  $(G, E_n)$ .

**Lemma 4.** In 2-fair energy games, player 1 has memoryless winning strategies.

We note that Lem. 4 is a corollary of Thm. 4. As 2-fair energy games are not determined,  $Win_2 \neq Q \setminus Win_1$ . We present a simple reduction to compute  $Win_2$  and resulting memory requirements of strategies as a corollary of its proof.

**Lemma 5.** Given a 2-fair energy game  $(G, E_f, E_n)$ , the set of winning nodes for player 2 is the same as the set of winning nodes for player 2 in the 2-fair mean-payoff game  $(G, E_f, MP_0^f)$ .

Proof. Take a player 2 strategy  $\pi$  that wins from q in the 2-fair mean-payoff game. Then, for any  $\tau \in \mathtt{plays}_\pi(G,q)$ ,  $\mathtt{avg}(\tau) = -\epsilon$  for some positive  $\epsilon$ . That is, there exists an infinite subsequence I of  $\mathbb N$  such that for all  $i \in I$ ,  $w(\tau[0;i]) < -i \cdot \epsilon$ . Thus, the energy of  $\tau$  drops unboundedly. For the opposite direction, due to the determinacy of mean-payoff games and Lem. 3, it suffices to show that for each positional player 1 strategy  $\sigma$ , there exists a player 2 strategy  $\pi$  such that  $\tau = \mathtt{play}_{\sigma,\pi}(G,q)$  has negative limit average weight. Let  $\sigma$  be a positional player 1 strategy and  $\pi$  a player 2 strategy that wins q in the 2-fair energy game. Consider  $\tau = \mathbf{p} \cdot \hat{\tau}$ , where  $\hat{\tau}$  is a tail of  $\tau$  consisting of infinitely often visited nodes and edges of  $\tau$ . Since the energy of  $\hat{\tau}$  drops unboundedly, there exists a negative cycle  $\mathbf{c}$  in  $\hat{\tau}$  that visits all nodes and (fair) edges of  $\hat{\tau}$ . Let  $\pi_{\sigma}$  be a player 2 strategy that mimics  $\pi$  until  $\mathbf{c}$  is seen, and then repeats  $\mathbf{c}$ . Then  $\rho = \mathtt{play}_{\sigma,\pi_{\sigma}}(G,q)$  has a tail  $\mathbf{c}^{\omega}$ , and since  $w(\mathbf{c}) < 0$ , we have  $\mathtt{avg}(\rho) < 0$ .

**Lemma 6.** Player 2 has finite memory winning strategies in 2-fair energy games.

### 5.3 1-fair Energy Games

In this section, we prove that 1-fair energy games are determined and using that introduce a gadget to reduce 1-fair energy games to regular energy games. The determinacy of 1-fair energy games does not follow from prior work on  $\omega$ -regular energy games [7] as the definition of Win<sub>2</sub> there uses Eq. (6) instead of Eq. (4). We establish determinacy by showing that, as in energy games, one can restrict the quantification over  $c \in \mathbb{N}$  to a finite set (namely to  $c \in [0, nW]$ ) in Eq. (6). That is, if there exists a player 2 strategy  $\pi$  that satisfies  $\forall \sigma \in \Sigma_1, \mathtt{play}_{\sigma,\pi}(G,q) \not\in \mathtt{En}_{nW}^f$ , then there exists a winning player 2 strategy  $\pi^{\mathtt{ext}}$  (winning against all c) as the energy level of plays conforming to this strategy  $\pi^{\mathtt{ext}}$  drops unboundedly. It then follows from the discussion in Sec. 5.1 that 1-fair energy games are determined. The proof of Thm. 5 can be found in App. D.

**Theorem 5.** 1-fair energy games are determined.

Next we will show that the gadget in Fig. 4 turns 1-fair energy games into regular energy games. The difference of this gadget from Fig. 2a mostly lies in the different treatment of the zero-weight-cycles. We first discuss briefly why such cycles are important, and then introduce this gadget.

The importance of 0-cycles in 1-fair energy games. It is known that the winning regions of a player in energy games and mean-payoff games (with

objective  $MP_0$ ) coincide [15]. This is not true for the 1-fair variants of these games, as player 1 may need infinite memory to win in 1-fair mean-payoff games (see Sec. 3 and 4.1), but not in 1-fair energy games. However, the equivalence of winning regions still holds in fair games without 0-cycles. Intuitively, if there are no 0-cycles, for a winning player 1 (player 2) strategy  $\sigma$  ( $\pi$ ) in the fair energy game, the energy of each play  $\tau$  that conforms with  $\sigma$  ( $\pi$ ) grows (drops) unboundedly, which makes  $\tau$  also winning in the fair mean-payoff game. For instance, in a 1-fair energy game with a player 1 node with a single fair edge with 0 weight is winning for player 1; but if we add another fair edge to this node with weight -1, it becomes losing for player 1. However, when we consider the graph under the mean-payoff objective, both nodes remain in Win<sub>1</sub>.

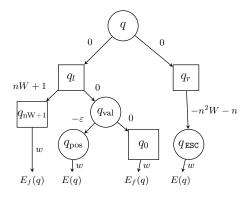


Fig. 4: 1-fair energy gadget where  $\epsilon = 1/(n+1)$ 

Therefore, a 1-fair energy gadget should distinguish between two kinds of nodes, A and B, both of which achieve mean-payoff value 0 but node A (in Win<sub>1</sub>) achieves value 0 due to a strategy that visits only 0-cycles while node B (in Win2) achieves it due to an infinite memory strategy that sees a 0-cycle more and more often. The middle branch of the gadget in Fig. 4 serves this purpose. The gadget matches Fig. 2a on the leftmost (br<sub>fair</sub>) and rightmost (br<sub>esc</sub>) branches, which still serve as the escape branches for player 1 and player 2, as in Sec. 4.1. The middle branch, called the value branch  $br_{val}^q$  distinguishes nodes of type A and B. Intuitively, if both players do not have escape strategies from q, then the value branch is taken. If player 1 can visit a positive cycle from q, then she wins from q since intuitively she can take this cycle enough times before visiting its fair edges. In this case, she chooses the left successor  $q_{pos}$  of  $q_{val}$ , which is called the positive value branch  $br_{pos}^q$  and visits the weight  $-\epsilon$ . The game is won by player 1 despite the weight  $-\epsilon$ , due to the positive cycle. On the other hand, if player 1 is winning from q but not by visiting a positive cycle, then she is winning as all the live edges lead to 0-cycles (Node A). In this case, player 1 takes the right successor  $q_0$  of  $q_{val}$ , which is called the zero value branch  $br_0^q$ , and wins. If neither of these cases holds for a node q (Node B), player 2 wins from both successors of the value branch.

**Theorem 6.** Let  $(G, E_f, MP_0^f)$  be a 1-fair energy game where G = (Q, E, w),  $w : E \to [-W, W]$ , and |Q| = n. Then there exists an energy game G' = (Q', E', w') where  $Q' \supseteq Q$ ,  $|Q'| \le 8n$  and  $w' : E' \to [-W', W']$  with  $W' = (n^2W + n)(n+1)$  such that  $Win_i(G) = Win_i(G') \cap Q$  for  $i \in \{1, 2\}$ .

We obtain the game G' in Thm. 6 by replacing each fair node in the 1-fair energy game G via the gadget in Fig. 4, as in Sec. 4.1. We note that the size of the game given in Thm. 6 is w.r.t. the equivalent energy game on integer weights, obtained by multiplying every weight in the game with  $1/\epsilon = n+1$ . The state-of-the-art algorithm for energy games [15] and Thm. 6 imply that 1-fair energy games can be solved in time  $O(n^4mw)$ . Furthermore, the proof reveals that player 1 has finite memory strategies (a local memory of  $log(n^3W + n)$  each node) and player 2 has positional strategies.

Unlike fair mean-payoff games, the determinacy of 1-fair energy games *does* not follow from Thm. 6. Instead, the proof hinges on the determinacy result itself, as detailed in App. D. Below, we outline the key ideas.

Recall that in the first direction of the proof of Thm. 1, we derive a player 1 strategy  $\sigma$  in G from a positional winning strategy  $\sigma'$  in G', reasoning about the tails  $\hat{\tau}$  of plays  $\tau \in \mathtt{plays}_{\sigma}(G)$ . We use the fact that in a mean-payoff (and similarly, in an energy) game, all cycles in  $G'_{\sigma'}$  are non-negative. This gives us that the cycles  $\mathbf{c}'$  using only simulation branches in  $G'_{\sigma'}$  have projections  $\mathbf{c} = \mathbf{c}'|_G$  with non-negative weight. In the 1-fair mean-payoff case, with infinite memory we let  $\sigma$  take these cycles increasingly often to ensure that the projection play  $\tau = \tau'|_G$  has non-negative mean-payoff.

Now, in 1-fair energy games, the cycles  $\mathbf{c}'$  in  $G'_{\sigma'}$  that visit a positive-value branch have positive weight projections  $\mathbf{c} = \mathbf{c}'|_Q$ , due to the  $-\epsilon$  weights. We refer to the choice of the positive-value branch,  $\sigma'(q_{\mathsf{pos}})$  as the 'preferred successor' of q that induces positive weight cycles in G. Thus, player 1 can adopt a strategy  $\sigma$  in G that repeatedly selects the preferred successors at fair nodes to accumulate sufficient positive weight before traversing the fair edges to satisfy fairness. As a result, the weight of every tail  $\hat{\tau}$  of all plays  $\tau \in \mathsf{plays}_{\sigma}(G)$  is monotonically increasing, and bounded below by some -d. However, unlike mean-payoff games, energy games require reasoning about prefixes of plays. Specifically, we need to ensure that the weight of every prefix of each play conforming to  $\sigma$  is bounded below by -c for some  $c \in \mathbb{N}$ . This is challenging, as the prefix lengths are unbounded, making the determinacy result crucial.

As these games are determined, player 2 wins from a node q if and only if it has a winning strategy  $\pi$  that wins against all initial credits c and all player 1 strategies (Eq. (4)). Therefore, the weight of all plays in  $\mathtt{plays}_{\pi}(G)$  drops unboundedly. Dually, a player 1 strategy  $\lambda$  is winning if all plays in  $\mathtt{plays}_{\lambda}(G)$  are fair with weights bounded below (not necessarily by the same c). Then for all  $\tau = \mathbf{u} \cdot \hat{\tau}$ , the length of the finite prefix  $\mathbf{u}$ , together with the lower bound -d of the tail  $\hat{\tau}$  naturally yields a lower bound of  $-d - W \cdot |\mathbf{u}|$  for the weight of  $\tau$ .

The proof of the second direction is similar to the case of 1-fair mean-payoff, where we construct a positional winning player 2 strategy  $\pi$  in G based on

a positional winning strategy  $\pi'$  in G'. This shows that memoryless strategies suffice for player 2. We give an alternative, simpler proof of this in App. D.

**Theorem 7.** In 1-fair energy games, player 1 has finite memory and player 2 has memoryless winning strategies.

## 6 Conclusion

In this work, we study the complexity of solving mean-payoff and energy games under strong transition fairness assumptions. We show that when combined with quantitative goals, fairness comes for free in the sense that the complexity of solving these games do not become computationally expensive. We discussed the determinacy and strategy complexity of these games and provided gadget-based techniques to solve them.

A possible future direction is to study the complexity of solving the doubly fair counterpart of these games, where both players are restricted by fairness constraints on their moves, similar to the extension of fair  $\omega$ -regular games in [34].

In the  $\omega$ -regular setting, a slight extension of strong transition fairness called group fairness is sufficient to make the problem NP-hard, even when combined with the reachability objective [46]. Group fairness is defined via tuples  $(G_1, E_1), \ldots, (G_k, E_k)$  where for each  $i \in [1, k], G_i$  is a subset of player 2 nodes, and  $E_i$  is a subset of edges such that the source node of each edge  $e \in E_i$  belongs to  $G_i$ . A play  $\tau$  is called group fair if, for each  $v \in V$  and  $i \in [1, k]$ , whenever  $v \in \text{Inf}(\tau) \cap G_i$ , there exists an  $e \in E_i$  that is taken infinitely often in  $\tau$ .

In a group fair game with an  $\omega$ -regular objective  $\varphi$  and tuples  $(G_1, E_1), \ldots$ ,  $(G_k, E_k)$  defining the group fairness, player 1 wins a play  $\tau$  if  $\tau$  satisfies  $\varphi$  or is not group fair. Being a subclass of Streett fairness (or, strong fairness), group fairness already unravels the full complexity of Streett fairness<sup>2</sup>. Since reachability can be expressed as a mean-payoff/energy objective, this NP-hardness result carries over to 2-group-fair<sup>3</sup> mean-payoff games, making these games NP-complete. 2-group-fair energy games again suffer from a lack of determinacy, since they subsume 2-fair energy games. Similarly, as the dual of reachability, the safety objective can be expressed as a mean-payoff/energy objective. This allows us to derive a coNP lower bound for 1-group-fair<sup>4</sup> mean-payoff and 1-group-fair energy games, making these games coNP-complete. However, the exploration of more general notions of fairness – such as other subclasses of Streett fairness – that could provide better complexity bounds or lead to elegant solution algorithms for qualitative or quantitative games remains a significant open area of research.

<sup>&</sup>lt;sup>2</sup> Note that in a game where fairness is inflicted on player 2 (as in the group fairness definition), the fairness condition is a subclass of Rabin objective rather than Streett. Thus, the derived complexity lower bound is NP rather than coNP.

<sup>&</sup>lt;sup>3</sup> By this, we indicate a game with group fairness where all  $G_i$  are player 2 nodes, as in the original definition of group fairness.

<sup>&</sup>lt;sup>4</sup> By this, we indicate a game with group fairness where all  $G_i$  are player 1 nodes. player 1 wins a play  $\tau$  of a 1-group-fair game with objective  $\varphi$  if  $\tau$  satisfies  $\varphi$  and is group fair.

#### References

- de Alfaro, L., Faella, M., Henzinger, T.A., Majumdar, R., Stoelinga, M.: Model checking discounted temporal properties. Theor. Comput. Sci. 345(1), 139–170 (2005). https://doi.org/10.1016/J.TCS.2005.07.033
- 2. Almagor, S., Boker, U., Kupferman, O.: Formalizing and reasoning about quality. In: Fomin, F.V., Freivalds, R., Kwiatkowska, M.Z., Peleg, D. (eds.) Automata, Languages, and Programming 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part II. Lecture Notes in Computer Science, vol. 7966, pp. 15–27. Springer (2013). https://doi.org/10.1007/978-3-642-39212-2\ 3
- 3. Almagor, S., Kuperberg, D., Kupferman, O.: The sensing cost of monitoring and synthesis. In: Harsha, P., Ramalingam, G. (eds.) 35th IARCS Annual Conference on Foundation of Software Technology and Theoretical Computer Science, FSTTCS 2015, December 16-18, 2015, Bangalore, India. LIPIcs, vol. 45, pp. 380–393. Schloss Dagstuhl Leibniz-Zentrum für Informatik (2015). https://doi.org/10.4230/LIPICS.FSTTCS.2015.380
- Almagor, S., Kupferman, O., Velner, Y.: Minimizing expected cost under hard boolean constraints, with applications to quantitative synthesis. In: Desharnais, J., Jagadeesan, R. (eds.) 27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada. LIPIcs, vol. 59, pp. 9:1–9:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2016). https://doi.org/10.4230/LIPICS.CONCUR.2016.9
- 5. Alur, R.: Principles of Cyber-Physical Systems. The MIT Press (2015)
- Aminof, B., Giacomo, G.D., Rubin, S.: Stochastic fairness and language-theoretic fairness in planning in nondeterministic domains. In: Beck, J.C., Buffet, O., Hoffmann, J., Karpas, E., Sohrabi, S. (eds.) Proceedings of the Thirtieth International Conference on Automated Planning and Scheduling, Nancy, France, October 26-30, 2020. pp. 20–28. AAAI Press (2020)
- Amram, G., Maoz, S., Pistiner, O., Ringert, J.O.: Energy mu-calculus: Symbolic fixed-point algorithms for omega-regular energy games. CoRR abs/2005.00641 (2020)
- 8. Asl, H.J., Uchibe, E.: Estimating cost function of expert players in differential games: A model-based method and its data-driven extension. Expert Systems with Applications **255**, 124687 (2024). https://doi.org/https://doi.org/10.1016/j.eswa.2024.124687
- 9. Baier, C., Katoen, J.P.: Principles of Model Checking (Representation and Mind Series). The MIT Press (2008)
- Banerjee, T., Majumdar, R., Mallik, K., Schmuck, A., Soudjani, S.: Fast symbolic algorithms for omega-regular games under strong transition fairness. TheoretiCS 2 (2023)
- Björklund, H., Sandberg, S., Vorobyov, S.G.: A combinatorial strongly subexponential strategy improvement algorithm for mean payoff games. In: Fiala, J., Koubek, V., Kratochvíl, J. (eds.) Mathematical Foundations of Computer Science 2004, 29th International Symposium, MFCS 2004, Prague, Czech Republic, August 22-27, 2004, Proceedings. Lecture Notes in Computer Science, vol. 3153, pp. 673–685. Springer (2004). https://doi.org/10.1007/978-3-540-28629-5\\_52
- Bloem, R., Chatterjee, K., Henzinger, T.A., Jobstmann, B.: Better quality in synthesis through quantitative objectives. In: Bouajjani, A., Maler, O. (eds.) Computer Aided Verification, 21st International Conference, CAV 2009, Grenoble,

- France, June 26 July 2, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5643, pp. 140–156. Springer (2009). https://doi.org/10.1007/978-3-642-02658-4\ 14
- Bohy, A., Bruyère, V., Raskin, J.: Symblicit algorithms for optimal strategy synthesis in monotonic markov decision processes. In: Chatterjee, K., Ehlers, R., Jha, S. (eds.) Proceedings 3rd Workshop on Synthesis, SYNT 2014, Vienna, Austria, July 23-24, 2014. EPTCS, vol. 157, pp. 51-67 (2014). https://doi.org/10.4204/EPTCS.157.8
- 14. Bouyer, P., Markey, N., Matteplackel, R.M.: Averaging in LTL. In: Baldan, P., Gorla, D. (eds.) CONCUR 2014 Concurrency Theory 25th International Conference, CONCUR 2014, Rome, Italy, September 2-5, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8704, pp. 266−280. Springer (2014). https://doi.org/10.1007/978-3-662-44584-6\ 19
- 15. Brim, L., Chaloupka, J., Doyen, L., Gentilini, R., Raskin, J.: Faster algorithms for mean-payoff games. Formal Methods Syst. Des. **38**(2), 97–118 (2011). https://doi.org/10.1007/S10703-010-0105-X
- Bruyère, V., Filiot, E., Randour, M., Raskin, J.: Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games. Inf. Comput. 254, 259–295 (2017). https://doi.org/10.1016/J.IC.2016.10.011
- Chakrabarti, A., de Alfaro, L., Henzinger, T.A., Stoelinga, M.: Resource interfaces. In: Alur, R., Lee, I. (eds.) Embedded Software, Third International Conference, EMSOFT 2003, Philadelphia, PA, USA, October 13-15, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2855, pp. 117–133. Springer (2003). https://doi.org/10.1007/978-3-540-45212-6\ 9
- Chatterjee, K., de Alfaro, L., Faella, M., Majumdar, R., Raman, V.: Code aware resource management. Formal Methods Syst. Des. 42(2), 146–174 (2013)
- Chatterjee, K., de Alfaro, L., Henzinger, T.A.: The complexity of stochastic rabin and streett games'. In: Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005, Lisbon, Portugal, July 11-15, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3580, pp. 878–890. Springer (2005). https://doi.org/10.1007/11523468 71
- 20. Chatterjee, K., Doyen, L.: Energy parity games. CoRR abs/1001.5183 (2010)
- Chatterjee, K., Doyen, L.: Energy and mean-payoff parity markov decision processes. In: Murlak, F., Sankowski, P. (eds.) Mathematical Foundations of Computer Science 2011 36th International Symposium, MFCS 2011, Warsaw, Poland, August 22-26, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6907, pp. 206–218. Springer (2011). https://doi.org/10.1007/978-3-642-22993-0\\_21
- 22. Chatterjee, K., Doyen, L.: Games and markov decision processes with mean-payoff parity and energy parity objectives. In: Kotásek, Z., Bouda, J., Cerná, I., Sekanina, L., Vojnar, T., Antos, D. (eds.) Mathematical and Engineering Methods in Computer Science 7th International Doctoral Workshop, MEMICS 2011, Lednice, Czech Republic, October 14-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7119, pp. 37–46. Springer (2011). https://doi.org/10.1007/978-3-642-25929-6\\_3
- 23. Chatterjee, K., Henzinger, T.A., Jurdzinski, M.: Mean-payoff parity games. In: 20th IEEE Symposium on Logic in Computer Science (LICS 2005), 26-29 June 2005, Chicago, IL, USA, Proceedings. pp. 178–187. IEEE Computer Society (2005). https://doi.org/10.1109/LICS.2005.26
- Chatterjee, K., Jurdzinski, M., Henzinger, T.A.: Simple stochastic parity games.
   In: CSL. Lecture Notes in Computer Science, vol. 2803, pp. 100–113. Springer (2003)

- Cimatti, A., Pistore, M., Roveri, M., Traverso, P.: Weak, strong, and strong cyclic planning via symbolic model checking. Artif. Intell. 147(1-2), 35–84 (2003)
- Clemente, L., Raskin, J.: Multidimensional beyond worst-case and almost-sure problems for mean-payoff objectives. In: 30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015. pp. 257–268.
   IEEE Computer Society (2015). https://doi.org/10.1109/LICS.2015.33
- 27. Cole, R., Hariharan, R., Paterson, M., Zwick, U.: Tighter lower bounds on the exact complexity of string matching. SIAM Journal on Computing **24**(1), 30–45 (1995). https://doi.org/10.1137/S0097539793245829
- 28. Comin, C., Rizzi, R.: Improved pseudo-polynomial bound for the value problem and optimal strategy synthesis in mean payoff games. Algorithmica **77**(4), 995–1021 (Apr 2017). https://doi.org/10.1007/s00453-016-0123-1
- 29. Daviaud, L., Jurdzinski, M., Lazic, R.: A pseudo-quasi-polynomial algorithm for mean-payoff parity games. In: Dawar, A., Grädel, E. (eds.) Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018. pp. 325–334. ACM (2018). https://doi.org/10.1145/3209108.3209162
- D'Ippolito, N., Rodríguez, N., Sardiña, S.: Fully observable non-deterministic planning as assumption-based reactive synthesis. J. Artif. Intell. Res. 61, 593–621 (2018)
- 31. Dorfman, D., Kaplan, H., Zwick, U.: A Faster Deterministic Exponential Time Algorithm for Energy Games and Mean Payoff Games. In: 46th International Colloquium on Automata, Languages, and Programming (ICALP 2019). Leibniz International Proceedings in Informatics (LIPIcs), vol. 132, pp. 114:1–114:14. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik (2019). https://doi.org/10.4230/LIPIcs.ICALP.2019.114
- 32. Ehrenfeucht, A., Mycielski, J.: Positional strategies for mean payoff games. Int. J. Game Theory 8(2), 109–113 (jun 1979). https://doi.org/10.1007/BF01768705
- 33. Francez, N.: Fairness. Springer, Berlin (1986)
- 34. Hausmann, D., Piterman, N., Saglam, I., Schmuck, A.: Fair ømega-regular games. In: FoSSaCS (1). Lecture Notes in Computer Science, vol. 14574, pp. 13–33. Springer (2024)
- 35. Hélouët, L., Markey, N., Raha, R.: Reachability games with relaxed energy constraints. In: Leroux, J., Raskin, J. (eds.) Proceedings Tenth International Symposium on Games, Automata, Logics, and Formal Verification, GandALF 2019, Bordeaux, France, 2-3rd September 2019. EPTCS, vol. 305, pp. 17–33 (2019). https://doi.org/10.4204/EPTCS.305.2
- 36. Kress-Gazit, H., Lahijanian, M., Raman, V.: Synthesis for robots: Guarantees and feedback for robot behavior. Annual Review of Control, Robotics, and Autonomous Systems 1(1), 211–236 (2018)
- 37. Majumdar, R., Mallik, K., Schmuck, A., Soudjani, S.: Symbolic control for stochastic systems via parity games. CoRR abs/2101.00834 (2021)
- 38. Martin, D.A.: Borel determinacy. Annals of Mathematics 102(2), 363-371 (1975)
- 39. Mehdipour, N., Althoff, M., Tebbens, R.D., Belta, C.: Formal methods to comply with rules of the road in autonomous driving: State of the art and grand challenges. Automatica 152, 110692 (2023). https://doi.org/https://doi.org/10.1016/j.automatica.2022.110692, https://www.sciencedirect.com/science/article/pii/S0005109822005568
- 40. Nilsson, P., Hussien, O., Balkan, A., Chen, Y., Ames, A.D., Grizzle, J.W., Ozay, N., Peng, H., Tabuada, P.: Correct-by-construction adaptive cruise control: Two

- approaches. IEEE Transactions on Control Systems Technology **24**(4), 1294–1307 (2016). https://doi.org/10.1109/TCST.2015.2501351
- 41. Nilsson, P., Ozay, N., Liu, J.: Augmented finite transition systems as abstractions for control synthesis. Discret. Event Dyn. Syst. **27**(2), 301–340 (2017)
- 42. Piterman, N., Pnueli, A.: Faster solutions of rabin and streett games. In: 21th IEEE Symposium on Logic in Computer Science (LICS 2006), 12-15 August 2006, Seattle, WA, USA, Proceedings. pp. 275–284. IEEE Computer Society (2006). https://doi.org/10.1109/LICS.2006.23
- Queille, J.P., Sifakis, J.: Fairness and related properties in transition systems

   a temporal logic to deal with fairness. Acta Inf. 19(3), 195–220 (1983).
   https://doi.org/10.1007/BF00265555
- 44. Saglam, I., Schmuck, A.: Solving odd-fair parity games. In: FSTTCS. LIPIcs, vol. 284, pp. 34:1–34:24. Schloss Dagstuhl Leibniz-Zentrum für Informatik (2023)
- 45. Scher, G., Kress-Gazit, H.: Warehouse automation in a day: From model to implementation with provable guarantees. In: 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE). pp. 280–287 (2020). https://doi.org/10.1109/CASE48305.2020.9217012
- Schmuck, A.K., Thejaswini, K.S., Sağlam, I., Nayak, S.P.: Solving two-player games under progress assumptions. In: Dimitrova, R., Lahav, O., Wolff, S. (eds.) Verification, Model Checking, and Abstract Interpretation. pp. 208–231. Springer Nature Switzerland (2024)
- 47. Streett, R.S.: Propositional dynamic logic of looping and converse. In: Proceedings of the Thirteenth Annual ACM Symposium on Theory of Computing. p. 375–383. STOC '81, Association for Computing Machinery (1981). https://doi.org/10.1145/800076.802492
- 48. Thejaswini, K.S.: Attractor decompositions for solving parity and Rabin games. Ph.D. thesis, Warwick (2023), https://wrap.warwick.ac.uk/id/eprint/187144/
- 49. Thistle, J.G., Malhamé, R.: Control of  $\omega$ -automata under state fairness assumptions. Systems & control letters **33**(4), 265–274 (1998)
- Velner, Y., Chatterjee, K., Doyen, L., Henzinger, T.A., Rabinovich, A., Raskin,
   J.: The complexity of multi-mean-payoff and multi-energy games. CoRR abs/1209.3234 (2012), http://arxiv.org/abs/1209.3234
- 51. Velner, Y., Chatterjee, K., Doyen, L., Henzinger, T.A., Rabinovich, A.M., Raskin, J.: The complexity of multi-mean-payoff and multi-energy games. Inf. Comput. **241**, 177–196 (2015). https://doi.org/10.1016/J.IC.2015.03.001
- 52. Wong, K.W., Ehlers, R., Kress-Gazit, H.: Resilient, provably-correct, and high-level robot behaviors. IEEE Transactions on Robotics **34**(4), 936–952 (2018). https://doi.org/10.1109/TRO.2018.2830353
- 53. Zwick, U., Paterson, M.: The complexity of mean payoff games on graphs. Theor. Comput. Sci.  $\bf 158 (1\&2)$ , 343-359 (1996). https://doi.org/10.1016/0304-3975(95)00188-3
- 54. Zwick, U., Paterson, M.: The complexity of mean payoff games on graphs. Theoretical Computer Science 158(1), 343–359 (1996). https://doi.org/https://doi.org/10.1016/0304-3975(95)00188-3

# A Formal construction of the player 1 gadget in Thm. 1

Formally, the 'gadget game' G' = (Q', E', w') with  $Q' = Q'_1 \uplus Q'_2$  is defined as follows:

- For every fair node  $q \in Q_f$ , we introduce two new player 1 nodes  $q_0, q_{ESC}$  and three new player 2 nodes  $q_l, q_r, q_{nW+1}$  in G'.

$$\begin{aligned} Q_1' = & Q_1 \cup \{q_0, q_{\text{ESC}} \mid q \in Q_f\} \\ Q_2' = & Q_2 \cup \{q_l, q_r, q_{\text{nW}+1} \mid q \in Q_f\} \end{aligned}$$

– With the introduction of the new nodes in the gadgets, we introduce new edges as follows:

$$E' = \{ qE \mid q \notin Q_f \}$$

$$\cup \{ (q, q_l), (q, q_r), (q_l, q_{nW+1}), (q_l, q_0), (q_r, q_{ESC}) \mid q \in Q_f \}$$

$$\cup \{ (q_{nW+1}, q') \mid q' \in E_f(q) \} \cup \{ (q_0, q'), (q_{ESC}, q') \mid q' \in E(q) \}$$

- We define the new weight function  $w': E' \to [-W', W']$  with  $W' = n^2W + n$  where n = |Q| as follows: for every  $q \notin Q_f, w'(e) = w(e)$ , for all  $e \in qE$ , and for every  $q \in Q_f$ ,

$$w'(q, q_l) = w'(q, q_r) = w'(q_l, q_0) = 0,$$

$$w'(q_l, q_{nW+1}) = nW + 1,$$

$$w'(q_r, q_{ESC}) = -n^2W - n,$$

$$w'(q_{nW+1}, q') = w(q, q') \text{ for } q' \in E_f(q),$$

$$w'(q_0, q') = w(q, q') \text{ for } q' \in E(q),$$

$$w'(q_{ESC}, q') = w(q, q') \text{ for } q' \in E(q)$$

## B Proof of Thm. 1 for 2-fair mean-payoff games

In this section, we will prove Thm. 1 for 2-fair mean-payoff games. We will do so via providing the correctness proof of the gadget in Fig. 2b, parallel to the proof in Sec. 4.1.

Although the proof is similar to that of 1-fair mean-payoff games, there is a substantial difference in strategy sizes. In 2-fair mean payoff games, the size of player 2 winning strategies are bounded by a local memory of  $\log(n^3+n^2+2n+2)$ , as opposed to 1-fair mean payoff games that has possibly infinite memory winning strategies for player 1. The reader curious about the reason of this difference in strategy sizes can jump to the second direction of the proof, and visit the definition of the winning player 2 strategy  $\pi$ , focusing on the bullet point c..

We start by giving a formal definition of the gadget game G' obtained by replacing each fair node in a 2-fair game with the 3-step gadget given in Fig. 2b.

Formally, the 'gadget game' G' = (Q', E', w') with  $Q' = Q'_1 \uplus Q'_2$  is defined as follows:

- For every fair node  $q \in Q_f$ , we introduce two new player 1 nodes  $q_0, q_{ESC}$  and three new player 2 nodes  $q_l, q_r, q_{nW+1}$  in G'.

$$\begin{aligned} Q_1' = & Q_1 \cup \{q_l, q_r, q_{\text{-nW-1}} \mid q \in Q_f\} \\ Q_2' = & Q_2 \cup \{q_0, q_{\text{ESC}} \mid q \in Q_f\} \end{aligned}$$

– With the introduction of the new nodes in the gadgets, we introduce new edges as follows:

$$E' = \{ qE \mid q \notin Q_f \}$$

$$\cup \{ (q, q_l), (q, q_r), (q_l, q_{-nW-1}), (q_l, q_0), (q_r, q_{ESC}) \mid q \in Q_f \}$$

$$\cup \{ (q_{-nW-1}, p^f), (q_0, p), (q_{ESC}, p) \mid q \in Q_f, p \in E(q) \text{ and } p^f \in E_f(q) \}$$

- We define the new weight function  $w': E' \to [-W', W']$  with  $W' = -n^2W - n$  where n = |Q| as follows: for every  $q \notin Q_f, w'(e) = w(e)$ , for all  $e \in qE$ , and for every  $q \in Q_f$ ,

$$\begin{split} w'(q,q_l) &= w'(q,q_r) = w'(q_l,q_0) = 0, \\ w'(q_l,q_{\text{-nW-1}}) &= -nW - 1, \\ w'(q_r,q_{\text{ESC}}) &= n^2W + n, \\ w'(q_{\text{-nW-1}},q') &= w(q,q') \text{ for } q' \in E_f(q), \\ w'(q_0,q') &= w(q,q') \text{ for } q' \in E(q), \\ w'(q_{\text{ESC}},q') &= w(q,q') \text{ for } q' \in E(q) \end{split}$$

For each  $q \in Q_f$ , we denote the leftmost branch  $q \to q_l \xrightarrow{-nW-1} q_{-nW-1}$  of the gadget as  $fair\ branch\ br^q_{\mathtt{fair}}$ , the middle branch  $q \to q_l \xrightarrow{0} q_0$  as the  $simulation\ branch\ br^q_{\mathtt{sim}}$  and the rightmost branch  $q \to q_r \xrightarrow{n^2W+n} q_{\mathtt{ESC}}$  as the  $escape\ branch\ br^q_{\mathtt{esc}}$ . The intuition behind the gadgets is idential to the 1-fair case.

We now prove the correctness of Thm. 1 for 2-fair mean-payoff games.

*Proof (Thm. 1 for 2-fair mean-payoff games).* We first show  $\text{Win}_1(G') \cap Q \subseteq \text{Win}_1(G)$ . Let  $\sigma'$  be a positional player 1 strategy that wins q in G' for the regular mean-payoff objective.

Consider the subgame arena  $G'_{\sigma'}$  of G'. All cycles in  $G'_{\sigma'}$  are non-negative; else player 2 can force to eventually only visit a negative cycle and therefore construct a play in  $G'_{\sigma'}$  with negative mean-payoff value.

We construct a positional player 1 strategy  $\sigma$  that wins q in G in a straightforward manner: for all  $q \in Q_1$ , we set  $\sigma(q) = \sigma'(q)$ .

For a play  $\tau = q^0 q^1 \dots \in \mathtt{plays}_{\sigma}(G)$ , we define its *extension*  $\tau'$  in G' inductively as follows: we start from with  $q^0$  and i = 0, and with increasing i,

- If  $q^i \notin Q_f$ , append  $\tau'$  with  $q^{i+1}$ ; Otherwise,
- If  $\operatorname{br}_{\mathtt{sim}}^{q^i}$  exists in  $G'_{\sigma'}$ , append  $\tau'$  with  $q^i_l \cdot q^i_0 \cdot q^{i+1}$ ;

- $\begin{array}{l} \text{ If } \operatorname{br}_{\mathtt{sim}}^{q^i} \text{ does not exist in } G'_{\sigma'} \colon \\ \bullet \text{ If } q^{i+1} = \sigma'(q^i_{\text{-nW-1}}), \text{ append } \tau' \text{ with the fair branch i.e. } q^i_l \cdot q^i_{\text{-nW-1}} \cdot q^{i+1}, \\ \bullet \text{ Otherwise, append it with the escape branch, i.e. } q^i_r \cdot q^i_{\text{ESC}} \cdot q^{i+1}. \end{array}$

Clearly,  $\tau'$  is a play conforming with  $\sigma'$ ; hence contains only non-negative cycles. Further, the restriction of  $\tau'$  to the nodes Q of G, denoted by  $\tau'|_Q$ , is exactly  $\tau$ . We define the tail of  $\tau$  as  $\hat{\tau}$  and its extension  $\hat{\tau}'$  similar to the proof for 1-fair mean-payoff. We denote the subgraph of  $G'_{\sigma'}$  restricted to the nodes and edges in  $\hat{\tau}'$  by  $G'_{\hat{\tau}'}$ .

Let  $\tau \in \operatorname{plays}_{\sigma}(G)$  be a play in G that conforms with  $\sigma$ . If  $\tau$  is not fair, it is automatically won by player 1. Therefore, assume that  $\tau$  is fair. We show that all the simple cycles in  $\hat{\tau}$  has non-negative weight. For this, we use the fact that its extension  $\tau'$  is a play in  $G'_{\sigma'}$  and hence all the cycles in  $G'_{\hat{\tau}'}$  have non-negative weight. Since the length of simple cycles are bounded by n, this gives us  $avg(\tau) > 0$ .

The main argument is the absence of fair branches in  $G'_{\hat{\tau}'}$ , introduced in Claim 2.

Claim 2 If  $\tau$  is fair, there exist no fair branches  $(br_{fair})$  in  $G'_{\hat{\tau}}$ .

Before proving the claim, we discuss how it implies that all simple cycles in  $\hat{\tau}$  are non-negative, resulting  $avg(\tau) \geq 0$ . From the definition of extension, absence of fair branches in  $G'_{\hat{\tau}'}$  implies that either (i) for all  $q \in Q_f$  only  $\mathsf{br}^q_{\mathsf{sim}}$  exists in  $G'_{\hat{\tau}'}$ , or (ii) for some  $q \in Q_f$ , only  $\operatorname{br}_{\operatorname{esc}}^q$  exists in  $\hat{\tau}'$ . By construction of  $\tau'$ , case (ii) above implies the existence of a fair node  $q \in Q_f$  that appears infinitely often in  $\tau$ but does not take all its fair outgoing edges, in particular  $q \to \sigma'(q_{-nW-1}) \in E_f(q)$ infinitely often. This makes  $\tau$  unfair, contradicting our assumption. In case of (i), since all simple cycles  $\mathbf{c}'$  in  $\hat{\tau}'$  are non-negative and none of the gadget nodes contribute any weight due to only  $br_{sim}$  existing in  $\hat{\tau}'$ , we conclude that the cycles  $\mathbf{c} = \mathbf{c}'|_Q$  in  $\hat{\tau}$  have non-negative weight. This implies that  $\operatorname{avg}(\tau) \geq 0$ .

By the above mentioned argument, proving Claim 2 concludes the first direction of the proof. We now prove the claim.

Proof of Claim 2. As discussed above, whenever  $\operatorname{br}_{\operatorname{esc}}^q$  is in  $G'_{\hat{\tau}'}$  for a node q,  $\mathsf{br}^q_{\mathsf{fair}}$  is also in it. Now we remove all the escape branches from  $G'_{\hat{\tau}'}$  and obtain the subgraph S. Note that S has no dead-ends, since for all q for which  $\mathbf{br}_{esc}^q$ 

got removed from  $G'_{\hat{\tau}'}$ ,  $\operatorname{br}^q_{\operatorname{fair}}$  was also in  $G'_{\hat{\tau}'}$ . For all nodes q in S, either  $\operatorname{br}^q_{\operatorname{fair}}$  or  $\operatorname{br}^q_{\operatorname{sim}}$  is in S. As a subgraph of  $G'_{\sigma'}$ , all cycles in S have non-negative weight. It is easy to see that a fair branch cannot lie on a simple cycle in S, because fair branches carry weight -nW-1 and the other edges come either from  $br_{sim}$  carrying weight 0, or from G carrying weight  $\leq W$ . Hence, a simple cycle containing a fair branch would have negative weight.

Now assume fair branches exist in  $\hat{\tau}'$ , but not on cycles of S. Then all cycles in S consist only of simulation branches. However, nodes with simulation branches have the same outgoing edges in  $G'_{\hat{\tau}'}$  and in S. Since every node in S is visited infinitely often in  $\tau$ ,  $\tau$  will eventually enter cycles of S and never leave them, since these nodes have the same outgoing edges in  $G'_{\hat{\pi}'}$ . This implies fair branches are not infinitely often visited in  $\tau$ , and concludes the proof of Claim 2.

This concludes the proof of the first direction. Now let us prove the other direction, i.e.  $Win_2(G') \cap Q \subseteq Win_2(G)$ .

Once more, let  $\pi'$  be a positional player 2 strategy that wins q in G'. We construct a player 2 strategy  $\pi$  that wins q in G. Consider the subgame arena  $G'_{\pi'}$  of G'. All cycles in  $G'_{\pi'}$  are negative; else player 1 can force to eventually only visit a non-negative cycle and therefore construct a play in  $G'_{\pi'}$  with non-negative mean-payoff value.

Using  $\pi'$ , we construct  $\pi$  in G as follows:

- a. If  $q' \notin Q_f$ , then set  $\pi$  to be positional at q' with  $\pi(q') = \pi'(q')$ . Note that this is well-defined since we do not use gadgets for nodes in  $Q \setminus Q_f$ .
- b. If  $\pi'(q') = q_r$  for some fair node  $q' \in Q_f$ , then set  $\pi$  to be positional at q' with  $\pi(q') = \pi'(q'_{ESC})$ . Intuitively this says if for a fair node q',  $\pi'$  asks to choose the escape branch of the gadget, then  $\pi$  follows  $\pi'$  to choose that branch and moves to the corresponding 'escape node' in G.
- c. If  $\pi'(q') = q_l$  for some fair node  $q' \in Q_f$ , then we set  $\pi$  to be a finite memory strategy that is played in rounds, using a local memory of size  $O(n^3W + n^2 + n + |E_f(q')| + 1)$  at q': consider an order on the fair successors of q' and keep a pointer j in the set  $E_f(q')$ . In round  $i \geq 0$ ,  $\pi$  choses the 'preferred successor'  $\pi'(q'_0)$  if  $i \mod n^3W + n^2 + n + 1 \neq 0$ . Otherwise, it takes the  $j^{\text{th}}$  (modulo  $|E_f(q')|$ ) fair edge of q' and increments j.

Now we prove that all  $\tau \in \mathtt{plays}_{\pi}(G)$  are winning for player 2 in G. In particular, we will show that  $\tau$  is fair and  $\mathtt{avg}(\tau) < 0$ .

For a play  $\tau = q^0 q^1 \dots \in \operatorname{plays}_{\pi}(G)$ , we construct its extension  $\tau'$  in G' inductively as follows: we start with  $q^0$  and i = 0 and with increasing i,

- If  $q^i \notin Q_f$ , append  $\tau'$  with  $q^{i+1}$ ;
- If  $\pi(H \cdot q^i)$  is defined using Item b. i.e.,  $\pi'(q^i) = q_r^i$  then append  $\tau'$  with  $\operatorname{br}_{\operatorname{esc}}^{q^i} \to q^{i+1}$ . In particular, we append  $\tau'$  with  $q_r^i \cdot q_{\operatorname{ESC}}^i \cdot q^{i+1}$ .
- If  $\pi(H \cdot q^i)$  is defined using Item c., then
  - if  $q^{i+1}=\pi'(q_0^i)$ , append  $\tau'$  with  $\operatorname{br}_{\mathtt{sim}}^{q^i}\to q^{i+1}$  i.e., with  $q_l^i\cdot q_0^i\cdot q^{i+1}$ ;
  - else, append  $\tau'$  with  $\operatorname{br}_{\mathsf{fair}}^{q^i} \to q^{i+1}$ , i.e.  $q_l^i \cdot q_{\mathsf{pW},1}^i \cdot q^{i+1}$ .

Observe that again,  $\tau'$  is a play in  $G'_{\pi'}$ ; hence it contains only negative cycles, and  $\tau'|_{O} = \tau$ . We define a tail  $\hat{\tau}'$  of  $\tau'$  as well as  $\hat{\tau}$  and  $G'_{\hat{\tau}'}$ , as before.

We first show that  $\tau$  is fair. It is sufficient to show that  $\hat{\tau}$  does not contain fair nodes p for which  $\pi(p)$  is defined via Item b., because for all the other fair nodes in  $\hat{\tau}$ ,  $\tau$  visits all their fair successors infinitely often (Item c.). Let p be a node for which  $\pi(p)$  is defined via Item b.. Then  $\pi'(p) = p_r$ . As  $\pi'$  is positional, only the escape branch of p exists in  $G'_{\hat{\tau}'}$ , carrying the weight  $n^2W + n$ . Since p is visited infinitely often, there is a (simple) cycle in  $G'_{\pi'}$  that passes through p, and thus sees the weight  $n^2W + n$ . In order to maintain a non-negative weight, the cycle needs to contain n-many fair branches (carrying weight -nW - 1), which is not possible since there are at most n-1 many fair branches in  $G'_{\pi'}$ , due to p taking its escape branch. With this we conclude that  $\tau$  is fair.

Now we prove that  $avg(\tau) < 0$ . First let us make an observation about the cycles in  $G'_{\pi'}$  that use only simulation branches.

Observation (sim-cycles) We call cycles  $\mathbf{c}'$  in  $G'_{\pi'}$  that do not contain any fair or escape branches sim-cycles. That is,  $\mathbf{c}'$  contains only simulation branches. Being part of  $G'_{\pi'}$ ,  $\mathbf{c}$  has negative weight. Since all the gadget branches in  $\mathbf{c}'$  carry 0 weight, the restriction  $\mathbf{c} = \mathbf{c}'|_{Q}$  is a negative weight cycle in G.

We saw that  $\hat{\tau}$  does not contain nodes defined using Item b.. I.e., for all fair nodes p in  $\hat{\tau}$ ,  $\pi$  is defined via Item c., and  $G'_{\hat{\tau}'}$  contains the branches  $\mathtt{br}^p_{\mathtt{fair}}$  and/or  $\mathtt{br}^p_{\mathtt{sim}}$ .

If  $\hat{\tau}$  does not contain any fair nodes,  $\hat{\tau} = \hat{\tau}'$ , and  $\operatorname{avg}(\tau) < 0$  easily follows. Now we assume  $\hat{\tau}$  contains at least one fair node. Then in order to show  $\operatorname{avg}(\tau) < 0$ , it is sufficient to show the existence of a number  $k \in \mathbb{N}$  and a positive rational number  $\epsilon$  such that every k-length infix of  $\hat{\tau}$  has average weight smaller than  $-\epsilon$ . We show this for  $k = n^3W + n^2 + n$  and  $\epsilon = 1/k$ .

Observe that every k-length infix  $\mathbf{u}'$  of  $\hat{\tau}'$  contains at least  $(n^2W+n+1)$  simple cycles, and the length of  $\mathbf{u}=\mathbf{u}'|_Q$  is at most k. Moreover, every fair branch is visited at most once in  $\mathbf{u}'$  (due to Item c.). Since  $\hat{\tau}'$  does not contain escape branches, every simple cycle in  $\hat{\tau}'$  is either a sim-cycle, or contains a fair branch. Since each fair branch is seen at most once in  $\mathbf{u}'$ , at most n of the cycles  $\hat{\tau}'$  contains are not sim-cycles. So,  $\hat{\tau}'$  contains at most n non-sim-cycles, and at least  $n^2W+1$  sim-cycles. Since the projection  $\mathbf{c}=\mathbf{c}'|_Q$  of a sim-cycle  $\mathbf{c}'$  to  $\mathbf{u}$  has weight at most -1; and the projection of a non-sim-cycle has weight at most nW;  $\mathbf{u}$  has weight at most -1. This concludes that every k-length infix of  $\hat{\tau}$  has averge weight at most -1/k.

This concludes the proof of the second direction and that of the theorem.

## C Proof of Thm. 4

*Proof.* Let q be a node in G. It is easy to see that, if player 1 wins from q in the regular energy game, then she also wins from q in the 2-fair energy game following the same strategy.

For the opposite side, we show that, if player 1 does not win from q in the regular energy game, then she does not win from q in the 2-fair energy game. We prove it by contradiction. Assume that player 1 does not win from q in  $(G, \operatorname{En})$  but wins from q in  $(G, E_f, \operatorname{En})$ . Then, there exists a finite initial credit c and player 1 strategy  $\sigma \in \mathcal{L}_1$  such that for all player 2 strategies  $\pi \in \mathcal{L}_2$ , the play  $\operatorname{play}_{\sigma,\pi}(G,q)$  either satisfies  $\operatorname{En}_c$  or it is not fair. On the other hand, positional determinacy of energy games implies that there exists a positional player 2 strategy  $\pi'$  in  $(G,\operatorname{En})$  that wins from q, i.e., all the plays conforming to  $\pi'$  violates energy. Consider the player 2 strategy  $\pi'_c$  in  $(G,E_f,\operatorname{En})$  that mimics  $\pi'$  in  $(G,E_f,\operatorname{En})$  until the accumulated energy of a play drops below 0 and then play fairly (satisfy the fairness condition). Note that, this is always possible since for a fixed initial credit c, the energy of a play drops below 0 after finitely many

steps (due to determinacy of energy games), and fairness is a prefix-independent objective. Clearly, the play  $\mathtt{play}_{\sigma,\pi'_c}(G,q)$  is fair and it does not satisfy  $\mathtt{En}_c$ , which leads to a contradiction. Also note that, this argument does not make the node q winning for player 2 in the 2-fair energy game, since the strategy  $\pi'_c$  depends on the finite initial credit c.

#### D Proofs of Sec. 5.3

#### D.1 Proof of Thm. 5

For this proof, we will need to define *strategy trees*.

Strategy Trees. The strategy tree  $\mathcal{T}_{\pi(q)}$  of a player i strategy  $\pi$  is obtained by unfolding the game  $(G, E_f)$  starting from node q, according to  $\pi$ . The nodes of the tree are prefixes  $u_1 \ldots u_k$  of  $\mathtt{plays}_{\pi}(G,q)$ . Therefore, each infinite branch corresponds to a play in  $\mathtt{plays}_{\pi}(G,q)$ . We will use nodes (branches) and the finite prefixes (plays) they represent, interchangably. Furthermore, we say the node  $u_1 \ldots u_k$  has weight  $w(u_1 \ldots u_k)$ .

In a 1-fair energy game a player 1 strategy  $\sigma$  is winning from q iff there exists a  $c \in \mathbb{N}$  such that all branches of  $\mathcal{T}_{\sigma(q)}$  are fair plays whose weights don't drop below c. A player 2 strategy  $\pi$  is winning from q if all branches of  $\mathcal{T}_{\pi(q)}$  are plays that are either unfair, or whose weights drop unboundedly (i.e. for each  $k \in \mathbb{N}$ , the weight of each fair branch in  $\mathcal{T}_{\pi(q)}$  drops below -k at some point). Observe that strategies and strategy trees for player i on game G have a 1-to-1 correspondance.

Proof (Thm. 5). Let  $q \in Q \setminus Win_1$ . Then, Eq. (3) and (4) holds. That is, for each  $c \in \mathbb{N}$ , there exists a player 2 strategy  $\pi_c$  that wins against all player 1 strategies. In particular for c = nW, there exists such a  $\pi_{nW}$ . Then the weight of all the fair branches of  $\mathcal{T}_{\pi_{nW}(q)}$  drop below -nW at some point. We will extend  $\mathcal{T}_{\pi_{nW}(q)}$  to obtain a winning player 2 strategy tree  $\mathcal{T}_{\pi^{\text{ext}}(q)}$ .

Let  $\rho$  be a fair branch in  $\mathcal{T}_{\pi_{nW}(q)}$ . Then we are guaranteed to find a prefix  $\mathbf{p} \cdot \mathbf{w}$  of  $\rho$  such that  $\mathbf{w}$  is a simple loop on a state p,  $w(\mathbf{p})$  is non-positive and  $w(\mathbf{w})$  is negative. To see this, observe that  $\rho$  has a prefix  $\mathbf{u} = u_1 \dots u_k$  with  $w(\mathbf{u}) < -nW$ . For each  $i \leq k$ , we denote  $u_1 \dots u_i$  with  $\mathbf{u}_i$ . Let  $\mathbf{u}_i$  be the longest prefix of  $\mathbf{u}$  with positive weight. Then,  $w(\mathbf{u}_{i+1}) \leq 0$ . Since  $w(u_i \dots u_k) < -nW$ , it contains a negative weight simple cycle. Let  $u_j \dots u_{j'}$  be the first such cycle with  $u_j = u_{j'} = p$ . Since  $j \geq i$ , clearly  $w(\mathbf{u}_j)$  is non-positive. Then, we set  $\mathbf{p} = \mathbf{u}_j$ , and  $\mathbf{w} = u_j \dots u_{j'}$ . Therefore, every fair branch  $\rho$  of  $\mathcal{T}_{\pi_{nW}(q)}$  has a unique signature,  $(\mathbf{p}_{\rho}, \mathbf{w}_{\rho})$ .

Now we are ready to build  $\mathcal{T}_{\pi^{\text{ext}}(q)}$ . We build the tree roughly by cutting each fair branch at  $\mathbf{p} \cdot \mathbf{w}$  and pasting the  $\mathbf{w}$  part from there on.

Formally, we start building  $\mathcal{T}_{\pi^{\text{ext}}(q)}$  through  $\mathcal{T}_{\pi_{nW}(q)}$ , branch by branch. For a finite prefix  $\mathbf{u}$  of each branch,  $\pi^{\text{ext}}$  prescribes the same successors to player 2 nodes as  $\mathcal{T}_{\pi_{nW}(q)}$ , until the finite prefix  $\mathbf{u}$  is equal to  $\mathbf{p} \cdot \mathbf{w}$  for a fair branch that has  $\mathbf{u}$  as a prefix. That is, the trees are identical until some  $\mathbf{p} \cdot \mathbf{w}$  is encountered. Then, for every player 2 node  $u \in \mathbf{w}$ ,  $\pi^{\text{ext}}$  prescribes the unique successor of u

in **w** to all u with history  $\mathbf{p} \cdot (\mathbf{w})^i \cdot \mathbf{w}'$  where  $i \geq 1$  and  $\mathbf{w}'$  a prefix of **w**. So,  $\pi^{\text{ext}}$  mimics the strategy of **w** for all plays with history  $\mathbf{p} \cdot (\mathbf{w})^i$ . For player 2 nodes with history  $\mathbf{p} \cdot (\mathbf{w})^i \cdot \mathbf{w}'$  where  $\mathbf{w}'$  is not a prefix of  $\mathbf{w}$ ;  $\pi^{\text{ext}}$  identifies these histories with  $\mathbf{p} \cdot \mathbf{w}'$  and prescribe such player 2 nodes, the same successors as nodes with history  $\mathbf{p} \cdot \mathbf{w}'$ . Note that, if  $\mathbf{p} \cdot \mathbf{w}'$  is a prefix of a fair branch  $\rho$  with signature  $(\mathbf{p}_{\rho}, \mathbf{w}_{\rho})$ , then  $\mathbf{p}$  is a prefix of  $\mathbf{p}_{\rho} \cdot \mathbf{w}_{\rho}$ . If this was not the case, while assigning the strategy to  $\mathbf{p}$  in  $\mathcal{T}_{\pi^{\text{ext}}(q)}$ , we would mimic the strategy of  $\rho$  wrt  $(\mathbf{p}_{\rho}, \mathbf{w}_{\rho})$ .

Let us look at branches  $\rho'$  in  $\mathcal{T}_{\pi^{\text{ext}}(q)}$ . Then  $\rho'$  satisfies one of the following three cases:

- (i) It eventually mimics a branch  $\rho$  in  $\mathcal{T}_{\pi_{nW}(q)}$ ,
- (ii) It has a tail of  $(\mathbf{w})^{\omega}$  where  $(\mathbf{p}, \mathbf{w})$  is the signature of a fair branch  $\rho$  in  $\mathcal{T}_{\pi_{\mathbf{v},\mathbf{w}}(\rho)}$ ,
- (iii) It visits cycles  $\mathbf{w}_1, \mathbf{w}_2, \ldots$  that belong to signatures  $(\mathbf{p}_1, \mathbf{w}_1), (\mathbf{p}_2, \mathbf{w}_2), \ldots$  of fair branches  $\rho_1, \rho_2, \ldots$  infinitely often, but it does not have a tail  $(\mathbf{w}_i)^{\omega}$  that cycles in only one of them. That is,  $\rho'$  is of the form  $\mathbf{w}'_1 \cdot (\mathbf{w}_1)^{j_1} \cdot \mathbf{w}'_2 \cdot (\mathbf{w}_2)^{j_2} \cdot \mathbf{w}'_3 \cdot (\mathbf{w}_3)^{j_3} \cdots$  for  $j_k \neq 0$  for all  $k \in \mathbb{N}$  and all  $\mathbf{w}'_i$  are minimal, in the sense that  $\mathbf{w}'_i$  does not contain  $\mathbf{w}_{i-1}$  as a prefix or  $\mathbf{w}_i$  as a suffix.

In case of Item (i)  $\rho'$  is either unfair, or it eventually mimics a play of the form  $\mathbf{p} \cdot (\mathbf{w})^{\omega}$  in  $\mathcal{T}_{\pi_{nW}(q)}$ . Since  $w(\mathbf{w})$  is negative, the weight of  $\rho'$  drops unboundedly. Similarly, in case of Item (ii), the weight of  $\rho'$  drops unboundedly.

For the rest of the proof, we will show that the weight of  $\rho'$  drops unboundedly in case of Item (iii). Recall that for all i,  $\mathbf{p}_i$  is a prefix of  $\mathbf{p}_{i+1} \cdot \mathbf{w}_{i+1}$ . Furthermore, the history  $\mathbf{p}_i \cdot (\mathbf{w}_i)^{j_i} \cdot \mathbf{w}'_{i+1}$  is identified with the history  $\mathbf{p}_i \cdot \mathbf{w}'_{i+1}$  in  $\pi^{\text{ext}}$ . The fact that the history  $\mathbf{p}_i \cdot (\mathbf{w}_i)^{j_i} \cdot \mathbf{w}'_{i+1}$  is continued by  $\mathbf{w}_{i+1}$  in  $\mathcal{T}_{\pi^{\text{ext}}}$  implies that  $\mathbf{p}_i \cdot \mathbf{w}'_{i+1}$  is equal to  $\mathbf{p}_{i+1} \cdot (\mathbf{w}_i)^k$  for some  $k \in \mathbb{N}$ .

In order to finish the proof, we will introduce the concept of shuffle-equivalence and prove Claim 3. Two finite plays  $\mathbf{u}_1$  and  $\mathbf{u}_2$  are called *shuffle-equivalent* iff they see the same cycles, the same number of times. Clearly, the weights of two shuffle-equivalent finite plays are equivalent.

We denote  $\rho'$  without the cycles  $\mathbf{w}_i$  as  $\rho^- = \mathbf{w}_1' \cdot \mathbf{w}_2' \cdots$  and denote the k-element prefix  $\mathbf{w}_1' \cdot \mathbf{w}_2' \cdots \mathbf{w}_k'$  of  $\rho^-$  with  $\rho_k^-$ . We show that  $\rho_k^-$  has non-positive weight for every k, which directly follows from the following claim.

Claim 3  $\rho_k^-$  is shuffle equivalent to  $\mathbf{p}_{k+1} \cdot (\mathbf{w}_1)^{z_1} \cdot (\mathbf{w}_2)^{z_2} \cdots (\mathbf{w}_{k+1})^{z_{k+1}}$  for some  $z_1, \ldots, z_k \in \mathbb{N}$ .

Proof (Claim 3). We will proceed by induction on k. The base case for k=1 follows from our previous observation that  $\mathbf{p}_1 \cdot \mathbf{w}_2'$  is equivalent to  $\mathbf{p}_2 \cdot (\mathbf{w}_2)^{z_2}$  for some  $z_2 \in \mathbb{N}$ . We assume the claim holds for k, and show it holds for k+1. By the inductive hypothesis,  $\rho_{k+1}^-$  is shuffle equivalent to  $\mathbf{p}_{k+1} \cdot (\mathbf{w}_1)^{z_1} \cdot (\mathbf{w}_2)^{z_2} \cdots (\mathbf{w}_{k+1})^{z_k} \cdot \mathbf{w}_{k+1}'$ . Then the claim follows the observation that  $\mathbf{p}_{k+1} \cdot \mathbf{w}_{k+1}'$  is of the form  $\mathbf{p}_{k+2} \cdot (\mathbf{w}_{k+2})^{z_{k+2}}$  for some  $z_{k+2} \in \mathbb{N}$ .

Claim 3 immediately implies that  $\rho_k^-$  has non-negative weight for each k, since  $\mathbf{p}_i$  and  $\mathbf{w}_i$  both have non-positive weight. This in turn implies that  $w(\rho_k') \leq -k$ 

where  $\rho'_k = \mathbf{w}'_1 \cdot (\mathbf{w}_1)^{j_1} \cdot \mathbf{w}'_2 \cdots \mathbf{w}'_k \cdot (\mathbf{w}_k)^{j_k}$  since all  $j_i$  are positive. This shows that the weight of  $\rho'$  drops unboundedly, which finishes the proof by revealing that  $\pi^{\text{ext}}$  is a winning player 2 strategy from q.

#### D.2 Proof of Thm. 6

Let  $(G, E_f, \operatorname{En}^f)$  be a 1-fair energy game. We construct G' by replacing every fair node in  $q \in Q_f$  in G, with its corresponding gadget presented in Fig. 4. Formally, the 'gadget game' G' = (Q', E', w') with  $Q' = Q'_1 \uplus Q'_2$  is defined as follows:

- For every fair node  $q \in Q_f$ , we introduce three new player 1 nodes  $q_{\text{val}}, q_{\text{pos}}, q_{\text{ESC}}$  and four new player 2 nodes  $q_l, q_r, q_{\text{nW}+1}, q_0$  in G'.

$$\begin{aligned} &Q_1' = Q_1 \cup \{q_{\text{val}}, q_{\text{pos}}, q_{\text{ESC}} \mid q \in Q_f\} \\ &Q_2' = Q_2 \cup \{q_l, q_r, q_{\text{nW}+1}, q_0 \mid q \in Q_f\} \end{aligned}$$

– With the introduction of the new nodes in the gadgets, we introduce new edges as follows:

$$E' = \{ qE \mid q \notin Q_f \}$$

$$\cup \{ (q, q_l), (q, q_r), (q_l, q_{\text{nW}+1}), (q_l, q_{\text{val}}), (q_{\text{val}}, q_{\text{pos}}), (q_{\text{val}}, q_0) (q_r, q_{\text{ESC}}) \mid q \in Q_f \}$$

$$\cup \{ (q_{\text{nW}+1}, q'), (q_0, q') \mid q' \in E_f(q) \} \cup \{ (q_{\text{pos}}, q'), (q_{\text{ESC}}, q') \mid q' \in E(q) \}$$

- We define the new weight function  $w': E' \to [-W', W'] \cup \{-\varepsilon\}$  with  $\varepsilon = \frac{1}{n+1}$  and  $W' = n^2W + n$  where n = |Q| as follows: for every  $q \notin Q_f$ , w'(e) = w(e), for all  $e \in qE$ , and for every  $q \in Q_f$ ,

$$\begin{split} w'(q,q_l) &= w'(q,q_r) = w'(q_l,q_{\mathtt{val}}) = w'(q_{\mathtt{val}},q_{\mathtt{0}}) = 0, \\ w'(q_l,q_{\mathtt{nW}+1}) &= nW+1, \\ w'(q_{\mathtt{val}},q_{\mathtt{pos}}) &= -\epsilon, \\ w'(q_r,q_{\mathtt{ESC}}) &= -n^2W-n, \\ w'(q_{\mathtt{nW}+1},q') &= w'(q_{\mathtt{0}}) = w(q,q') \text{ for } q' \in E_f(q), \\ w'(q_{\mathtt{pos}},q') &= w'(q_{\mathtt{ESC}},q') = w(q,q') \text{ for } q' \in E(q) \end{split}$$

For each  $q \in Q_f$ , we denote the leftmost branch  $q \to q_l \xrightarrow{nW+1} q_{nW+1}$  of the gadget as fair branch  $\operatorname{br}_{\mathtt{fair}}^q$ , the middle branch  $q \to q_{\mathtt{val}}$  as the value branch  $\operatorname{br}_{\mathtt{val}}^q$  and the rightmost branch  $q \to q_r \xrightarrow{-n^2W-n} q_{\mathtt{ESC}}$  as the escape branch  $\operatorname{br}_{\mathtt{esc}}^q$ . Furthermore, we denote the left branch  $q \to q_{\mathtt{val}} \xrightarrow{-\epsilon} q_{\mathtt{pos}}$  of the value branch as the positive-value branch  $\operatorname{br}_{\mathtt{pos}}^q$  and its right branch  $q \to q_{\mathtt{val}} \xrightarrow{0} q_0$  as the zero-value branch  $\operatorname{br}_0^q$ .

We note that the game graph as formally defined above, is equivalent to a game graph on integer weights obtained by multiplying every weight by  $\epsilon = \frac{1}{n+1}$ .

The size of the gadget game G' is stated in Thm. 6 is calculated with respect to the integer weights. For simplicity, in the formal definition and the proof we use the rational weights depicted on the figure.

We now formally prove the correctness of Thm. 6.

*Proof.* We first show that  $Win_1(G') \cap Q \subseteq Win_1(G)$  i.e., if player 1 wins the regular energy game from a node q in G', then player 1 also wins the fair energy game from q in G if q is in G.

Let  $\sigma'$  be a winning strategy for player 1 in G' for the regular energy objective starting from node q with  $q \in \text{Win}_1(G') \cap Q$ . Without loss of generality, let  $\sigma'$  be a positional strategy. We will construct a winning strategy  $\sigma$  for player 1 in G for the fair energy objective starting from node q.

Consider the subgame arena  $G'_{\sigma'}$  of G' with respect to the strategy  $\sigma'$ . Similar to the mean-payoff case, all cycles in  $G'_{\sigma'}$  are non-negative; else player 2 can force to eventually only visit a negative cycle and therefore construct a play in  $G'_{\sigma'}$  whose energy drops unboundedly. This leads to a contradiction since  $\sigma'$  is winning for player 1.

Using  $\sigma'$ , we construct  $\sigma$  in G as follows:

- a. If  $q' \notin Q_f$ , then set  $\sigma$  to be positional at q' with  $\sigma(q') = \sigma'(q')$ . Note that this is well-defined since we do not use gadgets for nodes in  $Q \setminus Q_f$ .
- b. If  $\sigma'(q') = q_r$  for some fair node  $q' \in Q_f$ , then set  $\sigma$  to be positional at q' with  $\sigma(q') = \sigma'(q'_{ESC})$ . Intuitively this says if for a fair node q',  $\sigma'$  asks to choose the escape branch of the gadget, then  $\sigma$  follows  $\sigma'$  to choose that branch and moves to the corresponding 'escape node' in G.
- c. If  $\sigma'(q') = q_l$  and  $\sigma(q_{\text{val}}) = q_{\text{pos}}$  for some fair node  $q' \in Q_f$ , then we set  $\sigma$  to be a finite memory strategy that is played in rounds, using a local memory of size  $\log(n^3W + |E_f(q)| + 1)$  at q': consider an order on the fair successors of q' and keep a pointer j in the set  $E_f(q')$ . In round  $i \geq 0$ ,  $\sigma$  chooses the 'preferred successor'  $\sigma'(q'_0)$  if  $i \mod n^3W + 1 \neq 0$ . Otherwise, it takes the  $j^{th}$  (modulo  $|E_f(q')|$ ) fair edge of q' and increments j.
- d. If  $\sigma'(q') = q_l$  and  $\sigma(q_{val}) = q_0$  for some fair node  $q' \in Q_f$ , then we let  $\sigma$  fix an order on the fair successors of q' as before, and loop through this set. Intuitively this choice of  $\sigma$  implies that all fair successors of q' lead to non-negative value cycles.

Recall that in 1-fair energy games, player 2 wins a node q iff it has a winning strategy  $\pi$  that wins against all initial credits c and all player 1 stragies (Eq. (4)). Therefore, the weight in all plays in  $\mathtt{plays}_{\pi}(G)$  drops unboundedly. This in turn implies that a player 1 strategy  $\lambda$  is winning if all plays in  $\mathtt{plays}_{\lambda}(G)$  are fair and have energies bounded from below (not necessarily by the same initial credit c).

Now we prove that all  $\sigma$  plays  $\tau \in \mathtt{plays}_{\sigma}(G)$  are winning for player 1 in G. In particular, we will show that (i)  $\tau$  is fair and (ii)  $\tau = \mathbf{w} \cdot \hat{\tau}$  for a finite prefix  $\mathbf{w}$  and an infinite suffix  $\hat{\tau}$  whose weight do not drop below  $-n^2W$ . Observe that (ii) implies that the weight of  $\tau$  is bounded from below.

For a play  $\tau = q^0 q^1 \dots \in \mathtt{plays}_{\sigma}(G)$ , we construct its extension  $\tau'$  in G' inductively as follows: we start with  $q^0$  and i = 0 and with increasing i,

- If  $q^i \notin Q_f$ , append  $\tau'$  with  $q^{i+1}$ ;
- If  $\sigma(H \cdot q^i)$  is defined using Item b. i.e.,  $\sigma'(q^i) = q_r^i$  then append  $\tau'$  with  $\begin{array}{l} \operatorname{br}_{\mathtt{esc}}^{q^i} \to q^{i+1}. \text{ In particular, we append } \tau' \text{ with } q^i_r \cdot q^i_{\mathtt{ESC}} \cdot q^{i+1}. \\ - \text{ If } \sigma(H \cdot q^i) \text{ is defined using Item c., } \sigma'(q^i) = q^i_l \text{ and } \sigma'(q^i_{\mathtt{val}}) = q^i_{\mathtt{pos}} \text{ then} \end{array}$
- - $\bullet \ \ \text{if} \ q^{i+1} = \sigma'(q^i_{\texttt{pos}}) \text{, append} \ \tau' \ \text{with} \ \texttt{br}^{q^i}_{\texttt{pos}} \rightarrow q^{i+1} \ \text{i.e., with} \ q^i_l \cdot q^i_{\texttt{pos}} \cdot q^{i+1};$
- else, append  $\tau'$  with  $\mathtt{br}_{\mathtt{fair}}^{q^i} \to q^{i+1}$ , i.e.  $q^i_l \cdot q^i_{\mathtt{nW}+1} \cdot q^{i+1}$ .

   If  $\sigma(H \cdot q^i)$  is defined using Item d.,  $\sigma'(q^i) = q^i_l$  and  $\sigma'(q^i_{\mathtt{val}}) = q^i_0$  then append  $\tau'$  with  $\operatorname{br}_0^{q^i} \to q^{i+1}$ , i.e., with  $q_l^i \cdot q_{\operatorname{val}}^i \cdot q_0^i \cdot q^{i+1}$ .

Clearly,  $\tau'$  is a play in  $G'_{\sigma'}$  and hence it contains only non-negative cycles. Furthermore, the restriction of  $\tau'$  to the nodes Q of G, denoted by  $\tau'|_Q$ , is exactly  $\tau$ .

Let a 'tail' of  $\tau$  denote the infinite suffix  $\hat{\tau}$  of  $\tau$  that contains only the nodes and edges visited infinitely often in  $\tau$ ; and let  $\hat{\tau}'$  be the extension of  $\tau$  in G'. We denote the subgraph of  $G'_{\sigma'}$  restricted to the nodes and edges in  $\hat{\tau}'$  by  $G'_{\hat{\tau}'}$ .

The proof of (i), i.e. that  $\tau$  is fair, is exactly the same as in 1-fair mean-payoff games. To show fairness of  $\tau$ , it is sufficient to show that  $\hat{\tau}$  does not contain fair nodes p for which  $\sigma(p)$  is defined as Item b. Let p be a node for which  $\sigma(p)$  is defined via Item b.. Then  $\sigma'(p) = p_r$ . As  $\sigma'$  is positional, among all the branches from the gadget of p, only  $\mathbf{br}_{\mathsf{esc}}^p$  exists in  $G'_{\sigma'}$  carrying the weight  $-n^2W-n$ . Since from our assumption p is visited infinitely often, there is a (simple) cycle in  $G'_{\sigma'}$  that passes through p, and thus sees the weight  $-n^2W - n$ . This cycle must be non-negative since it is a cycle in  $G'_{\sigma'}$ , but in order to counteract the negative weight  $-n^2W-n$ , the cycle needs to contain n-many gadget edges with weight nW + 1, which is not possible. Therefore,  $\tau$  is fair.

Now we prove that the weight of the prefixes  $\hat{\tau}_i$  of  $\hat{\tau}$  do not go below  $-n^2W$ . Similar to the proof of 1-fair mean-payoff, we make an observation about cycles in  $G_{\hat{\tau}'}$  that do not use left branches. Different from the mean-payoff case, we have 2 different kinds of simple cycles that do not see a left branch. One of them sees at least one positive value branch, and therefore collects weight  $-\epsilon$ . Since the cycle have non-negative value in G', and the gadget nodes contribute only negative weights; we projection of this cycle in G have positive weight. The second kind of cycles only see zero value branches. The projection of such cycles in G can have zero or positive weight.

Observation (positive-value and zero-value cycles) Let c' be a cycle in  $G'_{\sigma'}$  that does not contain any fair or escape branches. If  $\mathbf{c}'$  sees a positive value branch, we call it a positive-value cycle; else, we call it a zero-value cycle. For both positive- and zero-value cycles  $\mathbf{c}'$ , since the cycles have non-negative weight and the gadget nodes do not contribute positively to the weight,  $\mathbf{c}'|_Q$  has nonnegative weight. Furthermore in a positive-value cycle  $\mathbf{c}'$  the gadget nodes contribute negatively to the weight of the cycle. Therefore,  $w(\mathbf{c'}_Q) \geq 1$ .

We have shown above that  $\hat{\tau}$  contains no nodes for which  $\sigma$  has been defined via Item b.. Therefore, for all fair nodes p in  $\hat{\tau}$ ,  $\sigma$  is defined via Item c. or Item d., and  $G'_{\tau'}$  contains only the branches  $\operatorname{br}_{\operatorname{fair}}^p$  and/or  $\operatorname{br}_{\operatorname{val}}^p$ .

If no fair branches exist in  $\hat{\tau}'$ , then for no  $p \in \hat{\tau}$ ,  $\sigma$  is defined via Item d.. Therefore the only difference between  $\sigma'$  and  $\sigma$  are the zero-value branches which do not alter the weight of the game. The fact that the weight of  $\hat{\tau}$  is bounded below easily follows.

Otherwise, there exists a node  $p \in \hat{\tau}$  such that  $\mathbf{br}_{\mathtt{fair}}^p$  is in  $G'_{\hat{\tau}'}$ . We fix such a node p, and show in Claim 4 that for every infix  $\mathbf{u}'$  of  $\hat{\tau}'$  that corresponds to a simple cycle over  $\mathbf{br}_{\mathtt{fair}}^p$  in  $\hat{\tau}'$ ,  $\mathbf{u} = \mathbf{u}'|_Q$  has positive weight. Moreover, the weight of  $\mathbf{u}$  does not drop below  $-n^2W$  for any prefix. The proof of the claim concludes that the weight of  $\hat{\tau}$  is bounded below by  $-n^2W$ .

Claim 4 Let  $\mathbf{u}'$  be an infix of  $\hat{\tau}'$  which starts and ends at node p, the first branch of  $\mathbf{u}'$  as well as the branch following  $\mathbf{u}'$  in  $\hat{\tau}'$  are  $\operatorname{br}_{fair}^p$ . Then  $w(\mathbf{u}) > 0$ , for  $\mathbf{u} = \mathbf{u}'|_Q$ ; and  $w(\mathbf{u}_i) \geq -n^3W$  for every prefix  $\mathbf{u}_i$  of  $\mathbf{u}$ .

Proof (Claim 4). Observe that since  $\mathbf{u}'$  is a simple cycle on  $\mathbf{br}_{\mathbf{fair}}^p$ ; the fair branch of p is seen only once in  $\mathbf{u}'$ , i.e. as the first branch. However, the fair branches of the other nodes can repeat arbitrarily often in  $\mathbf{u}'$ . We will proceed by induction on the number of such nodes. We will show the following: If there are  $\leq (n-k)$ -many nodes whose fair branches are taken more than once in  $\mathbf{u}'$  for some  $k \in [1, n]$ , then  $w(\mathbf{u}) \geq (k-1)n^2W$  and  $w(\mathbf{u}_i) \geq -(n-k+1)n^2W$  for every prefix  $\mathbf{u}_i$  of  $\mathbf{u}$ .

Base case ( $\mathbf{k} = \mathbf{n}$ ): In this case, the fair branches of all nodes are seen at most once in  $\mathbf{u}'$ . That is, there are at most n-many fair branches in  $\mathbf{u}'$ , and the rest are positive- or zero-value branches. Consequently, in  $\mathbf{u}'$  there are at most n-many simple cycles that are neither positive- nor zero-value cycles. These are the only cycles whose projections to Q can have negative weight in  $\mathbf{u}$ ; and since they are simple cycles they can collect the minimal weight of  $-n^2W$ . Therefore, the weight of any infix of  $\mathbf{u}$  is lower bounded by  $-n^2W$ . Furthermore, we know that the positive-value branch  $\mathbf{br}_{pos}^p$  is visited exactly  $n^3W$ -many times in  $\mathbf{u}'$ . Therefore, there are at least  $n^3W$ -many positive-value cycles in  $\mathbf{u}'$ , collecting the minimum weight of  $n^3W$  in  $\mathbf{u}$ . Therefore,  $w(\mathbf{u}) \geq n^3W - n^2W = (n-1)n^2W$ .

**Inductive Hypothesis:** Assume the statement holds for  $i \in [k+1, n]$ , i.e. if  $\mathbf{u}'$  has at most n-i different fair branches repeating for some  $i \in [k+1, n]$ , then  $w(\mathbf{u}) \geq (i-1)n^2W$  and  $w(\mathbf{u}_i) \geq -(n-i+1)n^2W$  for any prefix  $\mathbf{u}_i$  of  $\mathbf{u}$ .

Inductive Step: We will show that the statement holds for k < n. WLOG we assume exactly (n - k)-many different fair branches repeat in  $\mathbf{u}'$ , since otherwise the desired inequality follows from the inductive hypothesis. We can find infices  $\mathbf{w^1}, \dots, \mathbf{w^t}$  of  $\mathbf{u}'$  for some  $t \in [1, n - k]$  such that each  $\mathbf{w^i}$  is a (not necessarily simple) cycle on the fair branch  $\mathbf{br_{fair}^{p_i}}$  of a unique node  $p_i$ . Note that the fair nodes of  $\mathbf{w^i}$  and  $\mathbf{w^j}$  are different for  $i \neq j$ . Furthermore, we can repeat this operation exhaustively such that if we remove the infices  $\mathbf{w^1}, \dots, \mathbf{w^t}$  from  $\mathbf{u}'$ , in the remaining sequence  $\mathbf{u}'$ \_there are no repeating fair branches.

Observe that  $\operatorname{br}_{\mathtt{fair}}^{p_i}$  can be visited multiple times in  $\mathbf{w^i}$ . Therefore, we view each  $\mathbf{w^i}$  as the non-empty concetanation  $\mathbf{w^i}_1 \cdot \mathbf{w^i}_2 \cdots \mathbf{w^i}_r$  where each  $\mathbf{w^i}_j$  is a simple cycle on  $\operatorname{br}_{\mathtt{fair}}^{p_i}$  for  $j \in [1, r]$ . Then the fair branches of at most (n - (k+1))-many nodes repeat in  $\mathbf{w^i}_j$ . By inductive hypothesis,  $w(\mathbf{w^i_j}|_Q) \geq kn^2W$ , and the

weight of  $w_i^i|_Q$  does not drop below  $-(n-k)n^2W$ . Consequently,  $w(\mathbf{w}^i|_Q) \geq$  $kn^2W$  and the weight of  $w^i|_Q$  does not drop below  $-(n-k)n^2W$ .

Recall that in  $\mathbf{u}'_{-}$  there are no repeating fair branches; therefore, there are at most n-many fair branches. Similar to the base case, there are at most n-many simple cycles  $\mathbf{c}'$  in  $\mathbf{u}'$  for which  $\mathbf{c} = \mathbf{c}'|_Q$  have negative weight. These cycles can collect the minimum weight of  $-n^2W$ . Due to our assumption  $k \neq n$ , there is at least one infix  $\mathbf{w^1}$  of  $\mathbf{u'}$ . Since the weight of  $\mathbf{w^1}|_Q$  is at least  $kn^2W$ ;  $w(\mathbf{u}) \geq$  $(k-1)n^2W$ ; and since the weight of  $\mathbf{w^1}|_Q$  does not drop below  $-(n-k)n^2W$ , the weight of **u** does not drop below  $-(n-k+1)n^2W$ .

This finishes the induction, and the proof of the claim.

This shows that  $\sigma$  is a winning player 1 strategy, thereby concluding the proof of the first direction. Now let us prove the other direction. In particular, we show that, if a node is winning for player 2 in G', then it is also winning for player 2 in G i.e.,  $q \in Win_2(G') \Rightarrow q \in Win_2(G)$ .

Analogous to the previous part, we consider a positional winning strategy  $\pi'$  of player 2 in G' and construct, this time, a positional winning strategy  $\pi$ for player 2 in G. Note that, none of the fair nodes in G belong to player 2 and hence the construction of  $\pi$  is quite straight forward: for all  $q \in Q_2$ , we set  $\pi(q) = \pi'(q)$ . We define  $G'_{\pi'}$  as the subgame restricted to the strategy  $\pi'$ , as before.

For a play  $\tau = q^0 q^1 \dots \in \operatorname{plays}_{\pi}(G)$ , we define its extension  $\tau'$  in G' inductively as follows: we start from with  $q^0$  and i=0, and with increasing i,

- If  $q^i \notin Q_f$ , append  $\tau'$  with  $q^{i+1}$ ; Otherwise,
- If the value branch does not exist in  $G'_{\pi'}$ :

   If  $q^{i+1} = \pi'(q^i_{\mathrm{nW}+1})$ , append  $\tau'$  with the fair branch i.e.  $q^i_l \cdot q^i_{\mathrm{nW}+1} \cdot q^{i+1}$ ,

   Otherwise, append it with the escape branch, i.e.  $q^i_r \cdot q^i_{\mathrm{ESC}} \cdot q^{i+1}$ .
- If the value branch exists in  $G'_{\pi'}$ :

  - If  $q^{i+1} = \pi'(q_0^i)$ , append  $\tau'$  with the zero-value branch i.e.  $q_l^i \cdot q_{\mathtt{val}}^i \cdot q_0^i \cdot q^{i+1}$ , Otherwise, append it with the positive-value branch, i.e.  $q_l^i \cdot q_{\mathtt{val}}^i \cdot q_{\mathtt{pos}}^i \cdot q^{i+1}$ .

We define the tail of  $\tau$  as  $\hat{\tau}$  and its extension  $\hat{\tau}'$ , analogous to the previous part. We denote the subgraph of  $G'_{\pi'}$  restricted to the nodes and edges in  $\hat{\tau}'$  by  $G'_{\hat{\tau}'}$ .

Let  $\tau \in \mathsf{plays}_{\pi}(G)$  be a play in G that conforms with  $\pi$ . If  $\tau$  is not fair, it is automatically won by player 2. Therefore, let us assume that  $\tau$  is fair. We will show that the weight of  $\tau$  drops unboundedly, i.e. for every  $c \in \mathbb{N}$  there exists a prefix  $\tau_i$  of  $\tau$  whose weight is below c. Therefore,  $\pi$  is a winning player 2 strategy.

We show that all simple cycles in  $\hat{\tau}$  have non-positive weight, and moreover, negative weight cycles are visited infinitely often. It will then follow directly that the weight of  $\tau$  drops unboundedly. To prove this, we will use the fact that its extension  $\tau'$  is a play in  $G'_{\pi'}$  and hence all the cycles in  $G'_{\hat{\tau}'}$  have negative weight as  $\pi'$  is winning for player 2 for the energy objective in G'.

Our main argument is the absence of fair branches in  $G'_{\hat{\tau}'}$ , which we introduce in Claim 5.

Claim 5 If  $\tau$  is fair, there exist no fair branches  $(\operatorname{br}_{fair})$  in  $G'_{\hat{\tau}'}$ .

The proof of the claim is idential to the proof of Claim 1. The replacement of simulation branches with value branches in the proof of Claim 1 is sufficient for it to serve as the proof of Claim 5. Note that the addition of  $-\epsilon$  edges into the weights accumulated from the gagdet branches does not alter the proof at all.

Now we show the absence of fair branches in  $G'_{\hat{\tau}'}$  implies that in  $\hat{\tau}$  (i) all simple cycles have non-positive weight, and (ii) a negative weight cycle is taken infinitely often. From the definition of extension, (I) for all  $q \in Q_f$  only the value branch  $\operatorname{br}_{\operatorname{val}}^q$  exists in  $G'_{\hat{\tau}'}$ , or (II) for some  $q \in Q_f$ , only the escape branch  $\operatorname{br}_{\operatorname{esc}}^q$  exists in  $\hat{\tau}'$ . By construction of  $\tau'$ , case (II) implies that there exists a fair node  $q \in Q_f$ that appears infinitely often in  $\tau$  but does not take all its fair outgoing edges, in particular  $q \to \pi'(q_{nW+1}) \in E_f(q)$  infinitely often. This implies that  $\tau$  is unfair, contradicting our assumption. Now assume (I) holds and let  $\mathbf{c}'$  be a simple cycle in  $G'_{\hat{\tau}'}$ . There exists at most n-many gadget branches in  $\mathbf{c}'$  and each one contributes weight  $-\epsilon$  or 0 to  $w(\mathbf{c}')$ . So, the minimum weight collected by the gadget branches in  $\mathbf{c}'$  is  $-n \cdot \epsilon > -1$ , due to our initial choice of  $\epsilon = 1/(n+1)$ . Since  $w(\mathbf{c}') < 0$ , the non-gadget nodes contribute non-positive weights to  $\mathbf{c}'$ ; that is  $w(\mathbf{c}) \leq 0$  for  $\mathbf{c} = \mathbf{c}'|_Q$ . Therefore, we conclude that all simple cycles in  $\hat{\tau}$  have non-positive weight, i.e. (i) holds. Moreover, since we assumed  $\hat{\tau}$  is fair, each q with  $\operatorname{br}_{\operatorname{val}}^q$  in  $G'_{\hat{\tau}'}$  will take its zero-value branch infinitely often; therefore,  $\operatorname{br}_0^q$  is in  $G'_{\hat{\tau}'}$ . Therefore, a zero-value cycle  $\mathbf{c}'$ , i.e. a cycle that does not see any other gadget branches but  $br_0$ , is guaranteed to exist in  $G'_{\hat{\tau}'}$ . Note that, if there are no value branches in  $G'_{\hat{\tau}'}$ , any cycle in  $G'_{\hat{\tau}'}$  is trivially a zero-value cycle. Since  $w(\mathbf{c}') < 0$ , this time  $w(\mathbf{c}) < 0$  as well. Lastly, since every cycle in  $G'_{\hat{\tau}'}$  is taken infinitely often in  $\hat{\tau}'$ , in particular any zero-value cycle  $\mathbf{c}'$  is taken infinitely often in  $\hat{\tau}'$ ; and therefore a negative weight cycle **c** is taken infinitely often in  $\hat{\tau}$ , i.e. (ii) holds.

With this, we conclude the proof of the second direction, and that of the theorem.

#### D.3 Alternative Proof of Second Part of Thm. 7

As in Lem. 3, the sufficiency of positional winning strategies for player 2 in 1-fair energy games follow as a corollary of the proof of Thm. 6. Here, using determinacy, we provide an alternative, simpler proof.

The proof follows from [50, Lemma 2] The idea is to show that if player 2 is losing from a vertex q with all positional strategies, it loses from q. Due to determinacy, this gives us that if player 2 is winning from q, it is winning with a positional strategy.

*Proof.* Let q be a player 2 node, and assume WLOG that it has 2 outgoing edges  $q \xrightarrow{w_1} q_1$  and  $q \xrightarrow{w_2} q_2$ . Assume both  $q_1$  and  $q_2$  are winning for player 1. Then, there exist minimal initial credits (due to Eq. (3))  $c_1$  and  $c_2$  with which player 1 wins from  $q_1$  and  $q_2$ , respectively. Then, the initial credit  $\max(c_1 - w_1, c_2 - w_2)$  is sufficient for player 1 to win from q.