Reachability Games with Relaxed Energy Constraints

Loïc Hélouët Nicolas Markey
Inria & CNRS & Univ. Rennes, France

Ritam Raha
Chennai Mathematical Institute, India

We study games with reachability objectives under energy constraints. We first prove that under strict energy constraints (either only lower-bound constraint or interval constraint), those games are LOGSPACE-equivalent to energy games with the same energy constraints but without reachability objective (i.e., for infinite runs). We then consider two kinds of relaxations of the upper-bound constraints (while keeping the lower-bound constraint strict): in the first one, called *weak upper bound*, the upper bound is *absorbing*, in the sense that it allows receiving more energy when the upper bound is already reached, but the extra energy will not be stored; in the second one, we allow for *temporary violations* of the upper bound, imposing limits on the number or on the amount of violations.

We prove that when considering weak upper bound, reachability objectives require memory, but can still be solved in polynomial-time for one-player arenas; we prove that they are in coNP in the two-player setting. Allowing for bounded violations makes the problem PSPACE-complete for one-player arenas and EXPTIME-complete for two players.

1 Introduction

Games on weighted graphs. Weighted games are a common way to formally address questions related to consumption, production and storage of resources: the arenas of such game are two-player turn-based games in which transitions carry positive or negative integers, representing the accumulation or consumption of resource. Various objectives have been considered for such arenas, such as optimizing the total or average amount of resources that have been collected along the play, or maintaining the total amount within given bounds. The latter kind of objectives, usually referred to as *energy objectives* [8, 3], has been widely studied in the untimed setting [9, 11, 13, 17, 21, 23, 26, 6, 4, 14], and to a lesser extent in the timed setting [2, 5]. As their name indicates, energy objectives can be used to model the evolution of the available energy in an autonomous system: besides achieving its tasks, the system has to take care of recharging batteries regularly enough so as to never run out of power. Energy objectives were also used to model moulding machines: such machines inject molten plastic into a mould, using pressure obtained by storing liquid in a tank [7]; the level of liquid has to be controlled in such a way that enough pressure is always available, but excessive pressure in the tank would reduce the service life of the valve.

Energy games impose strict constraints on the total amount of energy at all stages of the play. Two kinds of constraints have been mainly considered in the literature: lower-bound constraints (a.k.a. L-energy constraints) impose a strict lower bound (usually zero), but impose no upper bound; on the other hand, lower- and upper-bound constraints (a.k.a. LU-energy constraints) require that the energy level always remains within a bounded interval $[L;U]$. Finding strategies that realize L-energy objectives along infinite runs is in PTIME in the one-player setting, and in $\mathsf{NP} \cap \mathsf{coNP}$ for two players; for LU-energy objectives, it is respectively PSPACE-complete and EXPTIME-complete [3]. Some works have also considered the existence of an initial energy level for which a winning strategy exist [11].

In this paper, we focus on weighted games combining energy objectives together with reachability objectives. Our first result is the (expected) proof that L-energy games with or without reachability objectives are interreducible; the same holds for LU-energy games. We then focus on relaxations of

J. Leroux and J.-F. Raskin (Eds.): Tenth International Symposium
on Games, Automata, Logics, and Formal Verification (GandALF'19).
EPTCS 305, 2019, pp. 17–33, doi:10.4204/EPTCS.305.2

the energy constraints, in two different directions. In both cases, the lower bound remains unchanged, as it corresponds to running out of energy, which we always want to avoid. We thus only relax the upper-bound constraint. The first direction concerns *weak upper bounds*, already introduced in [3]: in that setting, hitting the upper bound is allowed, but there is no overload: trying to exceed the upper bound will simply maintain the energy level at this maximal level. Yet, a strict lower bound is still imposed. Following [3], we name these objectives LW-energy objectives. They could be used as a (simplified) model for batteries. When considered alone, LW-energy objectives are not much different from L-energy objectives, in the sense that the aim is to find a reachable *positive loop*. LW-energy games (with no other objectives besides maintaining the energy above L) are in PTIME for one-player games, and in NP $\cap$ coNP for two players [3]. When combining LW-energy and reachability objectives, we prove in this paper that the situation changes: different loops may have different effects on the energy level, and we have to keep track of the final energy level reached when iterating those loops.

We introduce and study a second way of relaxing upper bounds, which we call *soft upper bound*: it consists in allowing a limited number (or amount) of violations of the soft upper bound (possibly within an additional strong upper-bound): when modeling a pressure tank, the lower-bound constraint is strict (pressure should always be available) but the upper bound is soft (excessive pressure may be temporarily allowed if needed). We consider different kinds of restrictions (on the number or amount of violations), and prove that deciding whether Player 1 has a strategy to keep violations below a given bound is PSPACE-complete for one-player arenas, and EXPTIME-complete for two-player ones. We also provide algorithms to optimize violations of the soft upper bound under a given strict upper bound.

Related work. Quantitative games have been the focus of numerous research articles since the 1970s, with various kinds of objectives, such as ultimately optimizing the total payoff, mean-payoff [15, 27], or discounted sum [27, 1]. Energy objectives, which are a kind of safety objectives on the total payoff, were introduced in [8] and rediscovered in [3]. Several works have extended those works by combining quantitative conditions together, e.g. multi-dimensional energy conditions [17, 23] or conjunctions of energy- and mean-payoff objectives [11]. Combinations with qualitative objectives (e.g. reachability [10] or parity objectives [9, 12, 14]) were also considered. Similar objectives have been considered in slightly different settings e.g. Vector Addition Systems with States [25] and one-counter machines [19, 20].

2 Preliminaries

Definition 1 *A two-player arena is a 3-tuple $G = (Q_1, Q_2, E)$ where $Q = Q_1 \uplus Q_2$ is a set of states, $E \subseteq Q \times \mathbb{Z} \times Q$ is a set of weighted edges. For $q \in Q$, we let $qE = \{(q, w, q') \in E \mid w \in \mathbb{Z},\ q' \in Q\}$, which we assume is non-empty for any $q \in Q$. A one-player arena is a two-player arena where $Q_2 = \emptyset$.*

Consider a state $q_0 \in Q$. A *finite path* in an arena G from an initial state q_0 is an finite sequence of edges $\pi = (e_i)_{0 \leq i < n}$ such that for every $0 \leq i < n$, writing $e_i = (q_i, w_i, q'_i)$, it holds $q'_i = q_{i+1}$. Fix a path $\pi = (e_i)_{0 \leq i < n}$. Using the notations above, we write $|\pi|$ for the size n of π, $\hat{\pi}_i$ for the i-th state q_i of π (with the convention that $q_n = q'_{n-1}$), and $\mathsf{first}(\pi) = \hat{\pi}_0$ for its first state and $\mathsf{last}(\pi) = \hat{\pi}_n$ for its last state. The empty path is a special finite path from q_0; its length is zero, and q_0 is both its first and last state. Given two finite paths $\pi = (e_i)_{0 \leq i < n}$ and $\pi' = (e'_j)_{0 \leq j < n'}$ such that $\mathsf{last}(\pi) = \mathsf{first}(\pi')$, the concatenation $\pi \cdot \pi'$ is the finite path $(f_k)_{0 \leq k < n+n'}$ such that $f_k = e_k$ if $0 \leq k < n$ and $f_k = e'_{k-n}$ if $n \leq k < n + n'$. For $0 \leq k \leq n$, the k-th prefix of π is the finite path $\pi_{<k} = (e_i)_{0 \leq i < k}$. We write $\mathsf{FPaths}(G, q_0)$ for the set of finite paths in G issued from q_0 (we may omit to mention G in this notation when it is clear from the context). Infinite paths are defined analogously; we write $\mathsf{Paths}(G, q_0)$ for the set of infinite paths from q_0.

A *strategy* for Player 1 (resp. Player 2) from q_0 is a function $\sigma\colon \mathsf{FPaths}(q_0) \to E$ associating with any finite path π with $\mathsf{last}(\pi) \in Q_1$ (resp. $\mathsf{last}(\pi) \in Q_2$) an edge originating from $\mathsf{last}(\pi)$. A strategy is said *memoryless* when $\sigma(\pi) = \sigma(\pi')$ whenever $\mathsf{last}(\pi) = \mathsf{last}(\pi')$.

A finite path $\pi = (e_i)_{0\leq i<n}$ *conforms* to a strategy σ of Player 1 (resp. of Player 2) from q_0 if $\mathsf{first}(\pi) = q_0$ and for every $0 \leq k < n$, either $e_k = \sigma(\pi_{<k})$, or $\mathsf{last}(\pi_{<k}) \in Q_2$ (resp. $\mathsf{last}(\pi_{<k}) \in Q_1$). This is extended to infinite paths in the natural way. Given a strategy σ of Player 1 (resp. of Player 2) from q_0, the outcomes of σ is the set of all (finite or infinite) paths π issued from q_0 that conform to σ.

A game is a triple (G, q_0, O) where G is a two-player arena, q_0 is an initial state in Q, and $O \subseteq \mathsf{Paths}(G, q_0)$ is a set of infinite paths, also called *objective* (for Player 1). A strategy for Player 1 from q_0 is winning in (G, q_0, O) if its infinite outcomes all belong to O.

Given a set $R \subseteq Q$ of states, the reachability objective defined by R is the set of all paths containing some state in R, while the safety objective defined by R is the set of all infinite paths never visiting any state in R. In this paper, we also focus on *energy objectives* [8, 3], which we now define.

Definition 2 *Fix a finite-state arena $G = (Q_1, Q_2, E)$. Let $L \in \mathbb{Z}$. The L-energy arena associated with G is the infinite arena $G_{\mathsf{L}} = (C_1, C_2, T)$ where $C_1 = \{q_{err}\} \cup Q_1 \times [L; +\infty)$ and $C_2 = Q_2 \times [L; +\infty)$ are sets of* configurations*, and $T \subseteq C_1 \times \mathbb{Z} \times C_2$ is such that*

- *for any (q, l) and (q', l') in $Q \times [L; +\infty)$ and any $w \in \mathbb{Z}$, we have $((q, l), w, (q', l')) \in T$ if, and only if, $(q, w, q') \in E$ and $l' = l + w \geq L$; we also impose a loop $(q_{err}, 0, q_{err}) \in T$.*
- *for any $(q, l) \in Q \times [L; +\infty)$, we have $((q, l), w, q_{err}) \in T$ if, and only if, there exists a transition $(q, w, q') \in E$ such that $l + w < L$*

Similarly, given $L \in \mathbb{Z}$ and $U \in \mathbb{Z}$, the LU-energy arena associated with G is the finite-state arena $G_{\mathsf{LU}} = (C_1, C_2, T)$ where $C_1 = \{q_{err}\} \cup Q_1 \times [L; U]$ and $C_2 = Q_2 \times [L; U]$, and $T \subseteq C_1 \times \mathbb{Z} \times C_2$ is such that

- *for any (q, l) and (q', l') in $Q \times [L; U]$ and any $w \in \mathbb{Z}$, we have $((q, l), w, (q', l')) \in T$ if, and only if, $(q, w, q') \in E$ and $l' = l + w \in [L; U]$; we also impose a loop $(q_{err}, 0, q_{err}) \in T$.*
- *for any $(q, l) \in Q \times [L; U]$, we have $((q, l), w, q_{err}) \in T$ if, and only if, there is a transition $(q, w, q') \in E$ such that $l + w < L$ or $l + w > U$;*

Finally, given $L \in \mathbb{Z}$ and $W \in \mathbb{Z}$, the LW-energy arena associated with G is the finite-state arena $G_{\mathsf{LW}} = (C_1, C_2, T)$ where $C_1 = \{q_{err}\} \cup Q_1 \times [L; W]$ and $C_2 = Q_2 \times [L; W]$, and $T \subseteq C_1 \times \mathbb{Z} \times C_2$ is such that

- *for any (q, l) and (q', l') in $Q \times [L; W]$ and any $w \in \mathbb{Z}$, we have $((q, l), w, (q', l')) \in T$ if, and only if, $(q, w, q') \in E$ and $l' = \min(W, l + w) \geq L$; we also impose a loop $(q_{err}, 0, q_{err}) \in T$.*
- *for any $(q, l) \in Q \times [L; W]$, we have $((q, l), w, q_{err}) \in T$ if, and only if, there is a transition $(q, w, q') \in E$ such that $l + w < L$;*

An L-run (resp. LU-run, LW-run) ρ in G from q with initial energy level l is a path in G_{L} (resp G_{LU}, G_{LW}) from (q, l) never visiting q_{err}. With such a run $\rho = (t_i)_i$ in G, writing $t_i = ((q_i, l_i), w_i, (q'_i, l'_i))$, we associate the path $\pi = (e_i)_i$ such that $e_i = (q_i, w_i, q'_i)$. We define $\hat{\rho}_i = (q_i, l_i)$, corresponding to the i-th configuration along ρ, and $\widetilde{\rho}_i = l_i$, which we name the energy level in that configuration.

Similarly, a path π is said L-feasible (resp. LU-feasible, LW-feasible) from initial energy level L if there exists an L-run (resp. LU-run, LW-run) from $(\mathsf{first}(\pi), L)$ whose associated path is π. Notice that if such a run exists, it is unique (because paths are defined as sequences of transitions).

The L-energy (resp. LU-energy, LW-energy) objective is the set of infinite paths that are L-feasible (resp. LU-feasible, LW-feasible) (from initial energy level L). Similarly, given a target set $R \subseteq Q$, the L-energy- (resp. LU-energy-, LW-energy-) reachability objective is the set of L-feasible (resp. LU-feasible, LW-feasible) paths visiting R.

Remark 3 *Taking L as the initial energy level results in no loss of generality, since any energy level can be obtained by adding an initial transition from (q_0, L).*

In many cases, strong upper bounds are too strict, as many system do not break as soon as their maximal energy level is reached. Imposing a *weak* upper bound is a way to relax these constraints. We introduce another way to relax energy constraints, by allowing for (limited) violations of the upper bound: given two strict bounds L and U in $\mathbb{Z}$, a soft upper bound $S \in \mathbb{Z}$ with $L \leq S \leq U$, and an LU-run ρ, the set of violations along ρ is the set $\mathsf{V}(\rho) = \{i \in [0;|\rho|] \mid \widetilde{\rho}_i > S\}$ of positions along ρ where the energy level exceeds the soft upper bound S. There are many ways to quantify violations along a run. We consider three of them in this paper, namely the total number of violations, the maximal number of consecutive violations, and the sum of the violations. We thus define the following three quantities: $\#\mathsf{V}(\rho) = |\mathsf{V}(\rho)|$, $\overline{\#}\mathsf{V}(\rho) = \max\{i - j + 1 \mid \forall k \in [i, j].\ k \in \mathsf{V}(\rho)\}$, and $\Sigma\mathsf{V}(\rho) = \sum_{i \in \mathsf{V}(\rho)}(\widetilde{\rho}_i - S)$.

Figure 1 shows the evolution of $\#\mathsf{V}$ along a winning run in an $\mathsf{LSU}^{\#}$-energy game. One can notice that with a strong upper bound of 3, state q_t would not be reachable. On the other hand, if the strong upper bound is set to 6, then then there exists a run from q_0 to q_t, but that requires 3 violations of soft upper bound $S = 3$ (and the total amount of violations is 6).

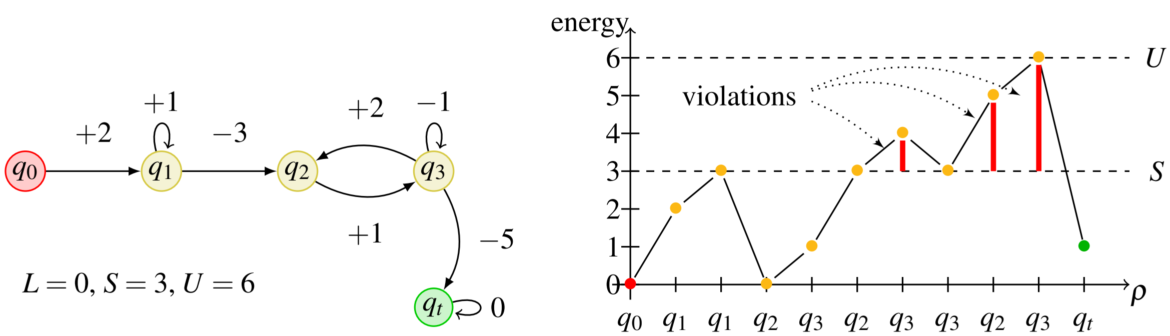

Figure 1: Energy level and $\#\mathsf{V}$ along a winning run in a $\mathsf{LSU}^{\#}$-energy reachability game.

Given three values $L \leq S \leq U$, and a threshold $V \in \mathbb{N}$, the $\mathsf{LSU}^{\#}$-energy (resp. $\mathsf{LSU}^{\overline{\#}}$-energy, LSU^{Σ}-energy) objective is the set of LU-feasible infinite paths π such that, along their associated runs ρ from (q_0, L), the number $\#\mathsf{V}(\rho)$ of violations (resp. maximal number of consecutive violations $\overline{\#}\mathsf{V}(\rho)$, sum $\Sigma\mathsf{V}(\rho)$ of violations) of the soft upper bound S is at most V. Similarly, for a set of states R, the $\mathsf{LSU}^{\#}$-energy-reachability (resp. $\mathsf{LSU}^{\overline{\#}}$-energy-reachability, LSU^{Σ}-energy-reachability) objective is the set of LU-feasible paths π reaching R such that along their associated run from (q_0, L), the number $\#\mathsf{V}(\rho)$ of violations (resp. maximal number of consecutive violations $\overline{\#}\mathsf{V}(\rho)$, sum $\Sigma\mathsf{V}(\rho)$ of violations) of the soft upper bound S is at most V.

We study the complexity of deciding the existence of a winning strategy for the objectives defined above, in both the one- and two-player settings. Further, for $\mathsf{LSU}^{\star}$-energy games (for $\star$ in $\{\#, \overline{\#}, \sum\}$), we also address the following problems:

- **bound existence:** Given L, S and V, decide if there exists a value $U \in \mathbb{Z}$ such that Player 1 wins the $\mathsf{LSU}^{\star}$-energy game;
- **minimization:** Given L and S, and a bound $V_{\max}$, find a value $U \in \mathbb{Z}$ such that Player 1 wins the game with the least possible violations less than $V_{\max}$, if any.

Table 1 summarizes known results, and the results obtained in this paper (where $\mathsf{LSU}^{\star}$-energy gathers all three energy constraints with violations). We furthermore show that the minimization problem for

LSU*-energy (reachability) games require algorithms that run in PSPACE in the one-player case, and in EXPTIME in the two-player case.

	Reachability		Infinite runs	
	1 player	2 players	1 player	2 players
L-energy	in PTIME (Thm. 4, [10])	in NP ∩ coNP (Thm. 4, [10])	in PTIME [3]	in NP ∩ coNP [3]
LU-energy	PSPACE-c. (Thm. 7)	EXPTIME-c. (Thm. 7)	PSPACE-c. [3]	EXPTIME-c. [3]
LW-energy	in PTIME (Thm. 22)	in coNP (Coro. 26)	in PTIME [3]	in NP ∩ coNP [3]
LSU*-energy	PSPACE-c. (Thm. 27)	EXPTIME-c. (Thm. 27)	PSPACE-c. (Thm. 27)	EXPTIME-c. (Thm. 27)
Bound existence	PSPACE-c. (Thm. 28)	EXPTIME-c. (Thm. 28)	PSPACE-c. (Thm. 28)	EXPTIME-c. (Thm. 28)
Minimization	PSPACE-c. (Thm. 29)	EXPTIME-c. (Thm. 29)	PSPACE-c. (Thm. 29)	EXPTIME-c. (Thm. 29)

Table 1: Summary of our results

3 Energy reachability games with strict bounds

In this section, we focus on the L-energy-reachability and LU-energy-reachability problems. We first prove that L-energy-reachability problems are inter-reducible with L-energy problems, which entails:

Theorem 4 *Two-player L-energy-reachability games are decidable in* NP ∩ coNP. *The one-player version is in* PTIME.

Remark 5 *It is worth noticing that these results are not a direct consequence of the results of [9] about energy parity games: in that paper, the authors focus on the* existence of an initial energy level *for which Player 1 has a winning strategy with energy-parity objectives (which encompass our energy-reachability objectives). When the answer is positive, they can compute the minimal initial energy level for which a winning strategy exists, but the (deterministic) algorithm runs in exponential time.*

Remark 6 *These results were already proven in [10]: for one-player arenas, the authors develop a* PTIME *algorithm, while they prove* LOGSPACE*-equivalence with L-energy games for the two-player setting (the result then follow from [3]). Our proof uses similar arguments as in the latter proof, but with the same full and direct reductions back and forth both for the one- and two-player cases.*

Proof. We prove that L-energy and L-energy-reachability are LOGSPACE-equivalent; our reductions are valid both for one- and two-player arenas. The result then follows from [3].

Intuitively, take a two-player L-energy game G, and build the arena G' as in Fig. 2: roughly, G' is obtained from G by increasing all weights by some small[1] positive value ε, and Player 1 is given the possibility to go to the target state, with a large negative cost $-\delta$, after each transition. If Player 1 wins in G, playing her winning strategy in G' will grow the energy level arbitrarily high, and will eventually allow her to take a transition to q_t. Conversely, if Player 1 wins in G', then she must be able to grow the energy to above δ; taking δ to be larger than the sum of all positive weights in G, Player 1 must have a strategy to force positive cycles before reaching q_t. If ε is small enough, any positive cycle in G' is a non-negative cycle in G, so that the repeating same strategy is winning in G.

The converse reduction is similar: this time, starting from a two-player L-energy-reachability game G, we build a two-player L-energy game G' by first restricting[2] to states from which Player 1 has a strategy

[1]The value of ε will generally be a rational, but it can be made integer by scaling up all constants.

[2]This may require to add a (losing) sink state, so that each state has an outgoing transition.

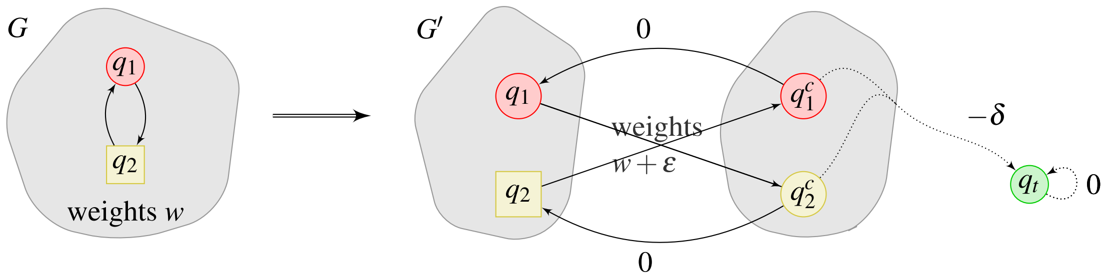

Figure 2: Schema of the reduction from L-energy to L-energy-reachability objectives

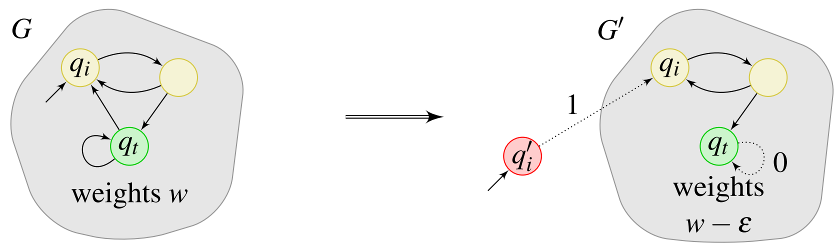

Figure 3: Schema of the reduction from L-energy-reachability to L-energy objectives

to reach q_t (with no quantitative constraints), removing the outgoing transitions of q_t and only keep a zero-self-loop, and substract a small positive ε to all other weights; to compensate with this, we set the initial energy level to 1. Then if Player 1 wins in G, then by taking ε small enough, the same strategy is winning in G'. Conversely, if she wins in G' (for L-energy objective), then some of the outcomes may reach q_t and some other will take non-negative cycles. When played in G, that strategy will reach q_t on some outcomes, and it will take positive cycles on some others; along the latter outcomes, the energy will grow arbitrarily high, and from some point on, Player 1 will simply have to switch to her attractor strategy in order to reach q_t. □

Similarly, for LU-energy-reachability objectives, we prove the same complexities as with classical LU-energy objectives:

Theorem 7 *One-player LU-energy-reachability games are* PSPACE*-complete. Two-player LU-energy-reachability games are* EXPTIME*-complete.*

Proof. Membership in EXPTIME is proven by considering the *expanded game* G_{LU}: it can be used to check reachability for both the one- and the two-player cases. For the one-player case, this can be achieved by proceeding on-the-fly, without explicitly building the expanded game; the resulting algorithm runs in PSPACE. For the two-player case, we solve reachability in that exponential-size game, which results in an EXPTIME algorithm.

For both the one- and the two-player settings, the hardness proofs for LU-energy objectives are readily adapted to LU-energy-reachability objectives, since they are based on reachability-like problems (reachability in bounded one-counter automata [18] and countdown games [22], respectively). □

4 Energy reachability games with weak upper bound

Finding a strategy that satisfies an LW-energy constraint along an infinite run is conceptually easy: it suffices to find a cycle that can be iterated once with a positive effect. It follows that memoryless strategies are enough, and the LW-energy problem was shown to be in PTIME for one-player arenas, and in NP ∩ coNP for two-player arenas [3].

The situation is different when we have a reachability condition: players may have to keep track of the exact energy level in order to find their way to the target state. Obviously, considering the expanded arena G_{LW}, we easily get exponential-time algorithms for LW-energy-reachability objectives. However, as proved below, in the one-player case, a PTIME algorithm exists.

Example 8 *Consider the one-player arena of Fig. 4, where the lower bound is $L = 0$, the weak-upper bound is $W = 5$, and the target state is q_t. Starting from q_0 with initial credit 0, we first have to move to q_1, and then iterate the positive cycle $\beta_1 = (q_1,+2,q_2)\cdot(q_2,-2,q3)\cdot(q_3,+1,q_1)$ three times, ending up in q_1 with energy level 3. We then take the cycle $\beta_2 = (q_1,+2,q_2)\cdot(q_2,-5,q4)\cdot(q_4,+5,q_1)$, which raises the energy level to 5 when we come back to q_1, so that we can reach q_t. Notice that β_1 has to be repeated three times before taking cycle β_2, and that repeating β_1 more than three times maintains the energy level at 4, which is not sufficient to reach q_t. This suggests that Player 1 needs memory and cannot rely on a single cycle to win LW-energy-reachability games.*

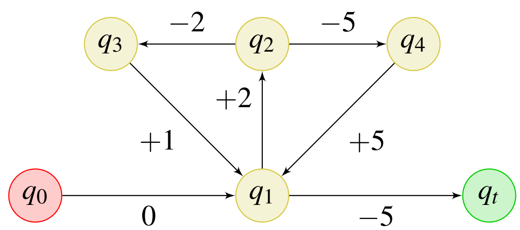

Figure 4: A one-player arena with LW-energy-reachability objective

This example shows that winning strategies for Player 1 may have to monitor the exact energy level all along the computation, thereby requiring exponential memory (assuming that all constants are encoded in binary; with unary encoding, the expanded game G_{LW} would have polynomial size and directly give a polynomial-time algorithm).

Proposition 9 *In LW-energy-reachability games, exponential memory may be necessary for Player 1 (assuming binary-encoded constants).*

Notice that this does not prevent from having PTIME algorithms: the strategy in Example 8 is not very involved, but it depends on the energy level (up to W).

From now on, and until Lemma 24, we consider the one-player case. In order to get a polynomial-time algorithm, we analyze cycles in the graph, and prove that a path witnessing LW-energy reachability can have a special form, which can be represented compactly using polynomial size. We begin with a series of simple lemmas.

Lemma 10 *Let π be a finite path in a one-player arena G. If $(q,u) \xrightarrow{\pi}_{\mathsf{LW}} (q',u')$, then for any $v \geq u$, $(q,v) \xrightarrow{\pi}_{\mathsf{LW}} (q',v')$ for some $v' \geq u'$.*

This lemma states that starting with higher energy level can only be beneficial. Notice that, even if we add condition $u' > u$ in the hypotheses of Lemma 10, it need not be the case that $v' > v$. In other terms, a sequence of transitions may have a positive effect on the energy level from some configuration, and a negative effect from some other configuration, due to the weak upper bound. Below, we prove a series of results related to this issue, and that will be useful for the rest of the proof. First, the effect of a given path (i.e., the net amount of energy that is harvested) decreases when the initial energy level increases:

Lemma 11 *Let π be a finite path in a one-player arena G, and consider two* LW*-runs $(q,u) \xrightarrow{\pi}_{\mathsf{LW}} (q',u')$ and $(q,v) \xrightarrow{\pi}_{\mathsf{LW}} (q',v')$ with $u \leq v$. Then $u'-u \geq v'-v$, and if the inequality is strict, then the energy level along the run $(q,v) \xrightarrow{\pi}_{\mathsf{LW}} (q',v')$ must have hit W.*

The next lemma is more precise about the effect of following a path when starting from the maximal energy level W:

Lemma 12 *Let π be a finite path in a one-player arena G, for which there is an* LW*-run $(q,u) \xrightarrow{\pi}_{\mathsf{LW}} (q',u')$. If u' is the maximal energy level along that run, then $(q,W) \xrightarrow{\pi}_{\mathsf{LW}} (q',W)$; if u is the maximal energy level along the run above, then $(q,W) \xrightarrow{\pi}_{\mathsf{LW}} (q',W+u'-u)$.*

From Lemma 10, it follows that any run witnessing LW-energy reachability can be assumed to contain no cycle with nonpositive effect. Formally:

Lemma 13 *Let π be a finite path in a one-player arena G. If $(q,u) \xrightarrow{\pi}_{\mathsf{LW}} (q',u')$ and π can be decomposed as $\pi_1 \cdot \pi_2 \cdot \pi_3$ in such a way that $(q,u) \xrightarrow{\pi_1}_{\mathsf{LW}} (s,v) \xrightarrow{\pi_2}_{\mathsf{LW}} (s,v') \xrightarrow{\pi_3}_{\mathsf{LW}} (q',u')$ with $v' \leq v$, then $(q,u) \xrightarrow{\pi_1 \cdot \pi_3}_{\mathsf{LW}} (q',u'')$ with $u'' \geq u'$.*

The following lemmas show that several occurrences of a cycle having positive effect along a path can be gathered together. This will be useful to prove the existence of a short path witnessing LW-energy reachability.

Lemma 14 *Let π be a finite path in a one-player arena G. If $(q,u) \xrightarrow{\pi}_{\mathsf{LW}} (q',u')$ with $u' > u$ and $(q,w) \xrightarrow{\pi}_{\mathsf{LW}} (q',w')$ with $w' > w$, then for any $u \leq v \leq w$, it holds that $(q,v) \xrightarrow{\pi}_{\mathsf{LW}} (q',v')$ with $v' > v$.*

Lemma 15 *Let π be a cycle on q such that $(q,u) \xrightarrow{\pi}_{\mathsf{LW}} (q,v)$ for some $u \leq v$. Then $(q,u) \xrightarrow{\pi^{W-L}}_{\mathsf{LW}} (q,v')$ for some $v' \geq v$, and $(q,v') \xrightarrow{\pi}_{\mathsf{LW}} (q,v')$.*

Fix a path π in G, and assume that some cycle φ appears (at least) twice along π: the first time from some configuration (q,u) to some configuration (q,u'), and the second time from (q,v) to (q,v'). First, we may assume that φ has length at most $|Q|$, since otherwise we can take an inner subcycle. We may also assume that $v > u'$, as otherwise we can apply Lemma 13 to get rid of the resulting nonpositive cycle between (q,u') and (q,v). For the same reason we may assume $u' > u$ and $v' > v$. As a consequence, by Lemmas 14 and 10, by repeatedly iterating φ from (q,u), we eventually reach some configuration (q,w) with $w \geq v'$, from which we can follow the suffix of π after the second occurrence of φ. It follows that all occurrences of φ along π can be grouped together, and we can restrict our attention to runs of the form $\alpha_1 \cdot \varphi_1^{n_1} \cdot \alpha_2 \cdot \varphi_2^{n_2} \cdots \varphi_k^{n_k} \cdot \alpha_{k+1}$ where the cycles φ_j are distinct, and have size at most $|Q|$, and the finite runs α_j are acyclic. Notice that each occurrence of any cycle φ_j can be assumed to have positive effect, and by Lemma 15, we may assume $n_j = W-L$ for all j.

While this allows us to only consider paths of a special form, this does not provide *short* witnesses, since there may be exponentially many cycles of length less than or equal to $|Q|$, and the witnessing run may need to iterate several cycles looping on the same state (see Example 8). In order to circumvent this problem, we have to show that all cycles need not be considered, and that one can compute the "useful" cycles efficiently. For this, we introduce *universal* cycles, which are cycles that can be iterated from any initial energy level (above L).

Definition 16 *Let G be a one-player arena, and q be a state of G. Let W be a weak-upper bound and $L \leq W$ be a lower bound. A* universal cycle *on q in G is a cycle φ with $\mathsf{first}(\varphi) = \mathsf{last}(\varphi) = q$ such that $(q,L) \xrightarrow{\varphi}_{\mathsf{LW}} (q, v_{\varphi,L})$ for some $v_{\varphi,L}$ (i.e., the energy level never drops below the lower bound L when following φ with initial energy level L). A universal cycle is* positive *if $v_{\varphi,L} > L$.*

When a cycle φ is iterated $W-L$ times in a row, then some universal cycle σ is also iterated $W-L-1$ times (by considering the state with minimal energy level along φ). As a consequence, iterating only universal cycles is enough: we may now only look for runs of the form $\beta_1 \cdot \sigma_1^{n_1} \cdot \beta_2 \cdot \sigma_2^{n_2} \cdots \sigma_k^{n_k} \cdot \beta_{k+1}$ where σ_j's are *universal* cycles of length at most $|Q|$. Now, assume that some state q admits two universal cycles σ and σ', and that both cycles appear along a given run π. Write h (resp. h') for the energy levels reached after iterating σ (resp. σ') $W-L$ times. We define a preorder on universal cycles of q by letting $\sigma \triangleright \sigma'$ when $h \geq h'$. Then if $\sigma \triangleright \sigma'$, each occurrence of σ' along π can be replaced with σ, yielding a run π' that still satisfies the LW-energy condition (and has the same first and last states). Generalizing this argument, each state that admits universal cycles has an optimal universal cycle of length at most $|Q|$, and it is enough to iterate only this universal cycle to find a path witnessing reachability. This provides us with a *small witness*, of the form $\gamma_1 \cdot \tau_1^{W-L} \cdot \gamma_2 \cdot \tau_2^{W-L} \cdots \tau_k^{W-L} \cdot \gamma_{k+1}$ where τ_j are optimal universal cycles of length at most $|Q|$ and γ_j are acyclic paths. Since it suffices to consider at most one universal cycle per state, we have $k \leq |Q|$. From this, we immediately derive an NP algorithm for solving LW-energy reachability for one-player arenas: it suffices to non-deterministically select each portion of the path, and compute that each portion is LW-feasible (notice that there is no need for checking universality nor optimality of cycles; those properties were only used to prove that small witnesses exist). Checking LW-feasibility requires computing the final energy level reached after iterating a cycle $W-L$ times; this can be performed by detecting the highest energy level along that cycle, and computing how much the energy level decreases from that point on until the end of the cycle. This provides us with a way of *accelerating* the computation of the effect of iterating cycles.

We now prove that optimal universal cycles of length at most $|Q|$ can be computed for a given state q_0. For this we unwind the graph from q_0 as a DAG of depth $|Q|$, so that it includes all cycles of length at most $|Q|$. We name the states of this DAG $[q',d]$, where q' is the name of a state of the arena and d is the depth of this state in the DAG (using square brackets to avoid confusion with configurations (q,l) where l is the energy level); hence there are transitions $([q',d],w,[q'',d+1])$ in the DAG as soon as there is a transition (q',w,q'') in the arena.

We then explore this DAG from its initial state $[q_0,0]$, looking for (paths corresponding to) universal cycles. Our aim is to keep track of all runs from $[q_0,0]$ to $[q',d]$ that are prefixes of universal cycles starting from q_0. Actually, we do not need to keep track of those runs explicitly, and it suffices for each such run to remember the following two values:

- the maximal energy level M that has been observed along the run so far (starting from energy level L, with weak upper bound W);
- the difference m between the maximal energy level M and the final energy level in $[q',d]$. Notice that $m \geq 0$, and that the final energy level in $[q',d]$ is $M-m$.

Example 17 *Figure 5 shows two universal cycles from q_0 in an LW-energy game with $L=0$ and $W=5$. The first cycle, going via q_2, ends with $M_1=5$ (reached in q_2) and $m_1=4$, thus with a final energy level of 1 (when starting from energy level 0); actually, iterating this cycle will not improve this final energy level. The second cycle, via q_4 and q_5, has a maximal energy level $M_2=4$ (reached in q_3) and ends with $m_2=1$. Hence, after one iteration of this cycle, one can end in state q_0 with energy level $W-m_2=4$.*

If we know the values (M,m) of some path from $[q_0,0]$) to $[q',d]$, we can decide if a given transition with weight w from $[q',d]$ to $[q'',d+1]$ can be taken (the resulting path can still be a prefix of a universal cycle if $M-m+w \geq L$), and how the values of M and m have to be updated: if $w>m$, the run will reach a new maximal energy level, and the new pair of values is $(\min(W;M-m+w),0)$; if $m+L-M \leq w \leq m$, then the transition can be taken: the new energy level $M-m+w$ will remain between L and M, and we

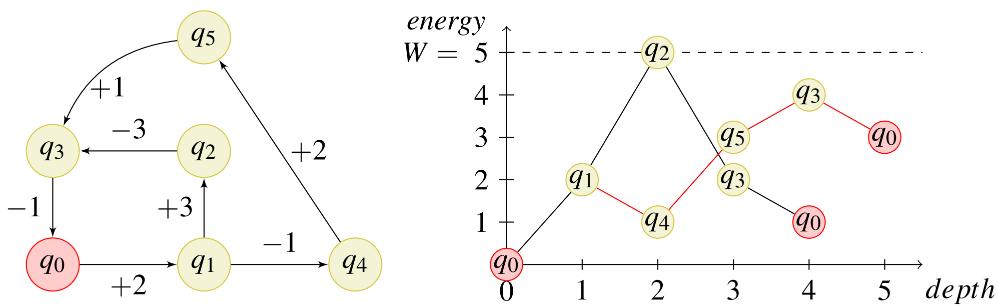

Figure 5: Two cycles with upper bound $W = 5$

update the pair of values to $(M, m-w)$; finally, if $w < m+L-M$, the energy level would go below L, and the resulting run would not be a prefix of a universal cycle.

Following these ideas, we inductively attach labels to the states of the DAG: initially, $[q_0, 0]$ is labelled with $(M = L, m = 0)$; then if a state $[q', d]$ is labelled with (M, m), and if there is a transition from $[q', d]$ to $[q'', d+1]$ with weight w:

- if $w > m$, then we label $[q'', d+1]$ with the pair $(\min(W; M-m+w), 0)$;
- if $m+L-M \leq w \leq m$, we label $[q'', d+1]$ with $(M, m-w)$.

The following lemma makes a link between runs in the DAG and labels computed by our algorithm:

Lemma 18 *Let $[q,d]$ be a state of the DAG, and M and m be two integers such that $0 \leq m \leq M-L$. Upon termination of this algorithm, state $[q,d]$ of the DAG is labelled with (M,m) if, and only if, there is an LW-run of length d from (q_0, L) to $(q, M-m)$ along which the energy level always remains in the interval $[L, M]$ and equals M at some point.*

Lemma 19 *Let $[q_0, d]$ be a state of the DAG, with $d > 0$. Let m be a nonnegative integer such that $L+m < W$. Upon termination of this algorithm, state $[q_0, d]$ is labelled with (M,m) for some $M > L+m$ if, and only if, there is a universal cycle φ on q_0 of length d such that $(q_0, L) \xrightarrow{\varphi^{W-L}}_{\mathsf{LW}} (q_0, W-m)$.*

Proof. First assume that $[q_0, d]$ is labelled with (M,m) for some M such that $M-m > L$. From Lemma 18, there is a cycle φ on q_0 of length d generating a run $(q_0, L) \xrightarrow{\varphi}_{\mathsf{LW}} (q_0, M-m)$ along which the energy level is within $[L, M]$. Then $M-m \geq L$, so that Lemma 15 applies: we then get $(q_0, L) \xrightarrow{\varphi^{W-L}}_{\mathsf{LW}} (q_0, E)$ with $(q_0, E) \xrightarrow{\varphi}_{\mathsf{LW}} (q_0, E)$ and $E \geq L$. Write $(p_i)_{0 \leq i < |\varphi|}$ for the sequence of weights along φ. Also write ρ for the run $(q_0, L) \xrightarrow{\varphi}_{\mathsf{LW}} (q_0, M-m)$, and σ for the run $(q_0, E) \xrightarrow{\varphi}_{\mathsf{LW}} (q_0, E)$.

As $L < M-m$, then by Lemma 11, it must be the case that energy level W is reached along σ. Write i_0 for the first position along ρ for which the energy level is M. Assume $\widetilde{\sigma}_{i_0} \neq W$: by Lemma 10, we must have $M = \widetilde{\rho}_{i_0} \leq \widetilde{\sigma}_{i_0} < W$. Then for all $k \geq i_0$, $\sum_{l=i_0}^{k} p_l \leq 0$. Since $\widetilde{\sigma}_{i_0} < W$, then also $\widetilde{\sigma}_k < W$ for all $k \geq i_0$. According to Lemma 11, energy level W is reached in σ, so there exists some $k_0 < i_0$ such that $\widetilde{\sigma}_{k_0} = W$. However, as i_0 is the index of the first maximal value in ρ, we have $\widetilde{\rho}_{k_0} < M$, and the energy level increases in run ρ between k_0 and i_0. So according to Lemma 12, we should have $\widetilde{\sigma}_{i_0} = W$, which raises a contradiction. Hence we proved $\widetilde{\sigma}_{i_0} = W$; applying the second result of Lemma 12, we get $E = W-m$.

Conversely, if there is a universal cycle φ satisfying the conditions of the lemma, then it must have positive effect when run from energy level L. Let F be such that $(q_0, L) \xrightarrow{\varphi}_{\mathsf{LW}} (q_0, F)$, and M be the maximal energy level encountered along the run $(q_0, L) \xrightarrow{\varphi}_{\mathsf{LW}} (q_0, F)$. By Lemma 18, state $[q_0, d]$ is labelled with (M, m') for some $m' \geq 0$ such that $F = M - m'$. By Lemma 15, we must have $(q_0, L) \xrightarrow{\varphi^{W-L}}_{\mathsf{LW}} (q_0, W - m')$. $\square$

The algorithm above computes optimal universal cycles, but it still runs in exponential time (in the worst case) since it may generate exponentially many different labels in each state $[q, d]$ (one per path from $[q_0, 0]$ to $[q, d]$). We now explain how to only generate polynomially-many pairs (M, m). This is based on the following partial order on labels: we let $(M, m) \preceq (M', m')$ whenever $M - m \leq M' - m'$ and $m' \leq m$. Notice in particular that

- if $M = M'$, then $(M, m) \preceq (M', m')$ if, and only if, $m' \leq m$;
- if $m = m'$, then $(M, m) \preceq (M', m')$ if, and only if, $M \leq M'$.

The following lemma entails that it suffices to store maximal labels w.r.t. $\preceq$:

Lemma 20 *Consider two paths π and π' such that $\mathsf{first}(\pi) = \mathsf{first}(\pi')$ and $\mathsf{last}(\pi) = \mathsf{last}(\pi')$, and with respective values (M, m) and (M', m') such that $(M, m) \preceq (M', m')$. If π is a prefix of a universal cycle φ, then π' is a prefix of a universal cycle φ' with $\varphi' \triangleright \varphi$.*

It remains to prove that by keeping only maximal labels, we only store a polynomial number of labels:

Lemma 21 *If the algorithm labelling the DAG only keeps maximal labels (for $\preceq$), then it runs in polynomial time.*

Proof. We prove that, when attaching to each node $[q, d]$ of the DAG only the maximal labels (w.r.t $\preceq$) reached for a path of length d ending in state q, the number of values for the first component of the different labels that appear at depth $d > 0$ in the DAG is at most $d \cdot |Q|$. Since it only stores optimal labels, our algorithm will never associate to a state $[q, d]$ two labels having the same value on their first component. So, any state at depth d will have at most $d \cdot |Q|$ labels.

So we prove, by induction on d, that the number of different values for the first component among the labels appearing at depth $d > 0$ is at most $d \cdot |Q|$. This is true for $d = 1$ since the initial state $(q, 0)$ only contains $(M = 0, m = 0)$, and each transition with nonnegative weight w will create one new label $(w, 0)$ (transitions with negative weight are not prefixes of universal cycles). Now, since all those labels have value 0 as their second component, each state $[q, 1]$ in the DAG will be attached at most one label. Hence, the total number of labels (and the total number of different values for their first component) is at most $|Q|$ at depth 1 in the DAG.

Now, assume that labels appearing at depth $d > 1$ are all drawn from a set of labels $\{(M_i, m_i) \mid 1 \leq i \leq n\}$ in which the number of different values of M_i is at most $d \cdot |Q|$. Consider a state $[q', d]$, labelled with $\{(M_i, m_i) \mid 1 \leq i \leq n_{q',d}\}$ (even if it means reindexing the labels). Pick a transition from $[q', d]$ to $[q'', d+1]$, with weight w. For each pair (M_i, m_i) associated with $[q', d]$, it creates a new label in $[q'', d+1]$, which is

- either $(\min(W; M_i - m_i + w), 0)$ if $m_i < w$ (maximal energy level increases);
- or $(M_i, m_i - w)$ if $m_i + L - M_i \leq w \leq m_i$ (maximal energy level in unchanged).

Now, for a state $(q'', d+1)$, the set of labels created by all incoming transitions can be grouped as follows:

- labels having zero as their second component; among those, our algorithm only stores the one with maximal first component, as $(M_i, 0) \preceq (M_j, 0)$ as soon as $M_i \leq M_j$;
- for each M_i appearing at depth d, labels having M_i as their first component; again, we only keep the one with minimal second component, as $(M, m_i) \preceq (M, m_j)$ when $m_j \leq m_i$.

Last, for this state $[q'', d+1]$, we keep at most one label for each distinct value among the first components M_i of labels appearing at depth d, and possibly one extra label with second value 0. In other terms, at depth $d+1$ the values that appear as first component of labels are obtained from values at depth d, plus possibly one per state; Hence, at depth $d+1$, there exists at most $(d+1)\cdot|Q|$ labels, which completes the proof of the induction step. □

Using the algorithm above, we can compute, for each state q of the original arena, the smallest value m_q for which there exists a universal cycle on q that, when iterated sufficiently many times, leads to configuration $(q, W-m_q)$. Since universal cycles can be iterated from any energy level, if q is reachable, then it is reachable with energy level $W-m_q$. We make this explicit by adding to our arena a special self-loop on q, labelled with $\mathsf{set}(W-m_q)$, which sets the energy level to $W-m_q$ (in the same way as *recharge transitions* of [16]).

In the resulting arena, we can restrict to paths of the form $\gamma_1 \cdot \nu_1 \cdot \gamma_2 \cdot \nu_2 \cdots \nu_k \cdot \gamma_{k+1}$, where ν_i are newly added transitions labelled with $\mathsf{set}(W-m)$, and γ_i are acyclic paths. Such paths have length at most $(|Q|+1)^2$. We can then inductively compute the maximal energy level that can be reached (under our LW-energy constraint) in any state after paths of length less than or equal to $(|Q|+1)^2$. This can be performed by unwinding (as a DAG) the modified arena from the source state q_0 up to depth $(|Q|+1)^2$, and labelling the states of this DAG by the maximal energy level with which that state can be reached from (q_0, L); this is achieved in a way similar to our algorithm for computing the effect of universal cycles, but this time only keeping the maximal energy level that can be reached (under LW-energy constraint). As there are at most $|Q|$ states per level in this DAG of depth at most $(|Q|+1)^2$, we get:

Theorem 22 *The existence of a winning path in one-player LW-energy-reachability games can be decided in* PTIME.

Example 23 *Consider the one-player arena of Fig. 6. We assume $L=0$, and fix an even weak upper bound W. The state s has $W/2$ disjoint cycles: for each odd integer i in $[0;W-1]$, the cycle c_i is made of three consecutive edges with weights $-i$, $+W$ and $-W+i+1$. Similarly, the state s' has $W/2$ disjoint cycles: for even integers i in $[0;W-1]$, the cycle c'_i has weights $-i$, $+W$ and $-W+i+1$. Finally, there are: two sequences of k edges of weight 0 from s to s' and from s' to s; an edge from the initial state to s of weight 1, and from s' to target state q_t of weight $-W$. The total number of states then is $2W+2k+2$.*

In order to go from the initial state q_0, with energy level 0, to the final state q_t, we have to first take the cycle c_1 (with weights -1, $+W$, $-W+2$) on s (no other cycles c_i can be taken). We then reach configuration $(s,2)$. Iterating c_1 has no effect, and the only next interesting cycle is c_2, for which we have to go to s'. After running c_2, we end up in $(s',3)$. Again, iterating c_2 has no effect, and we go back to s, take c_3, and so on. We have to take each cycle c_i (at least) once, and take the sequences of k edges between s and s' $W/2$ times each. In the end, we have a run of length $3W+Wk+2$.

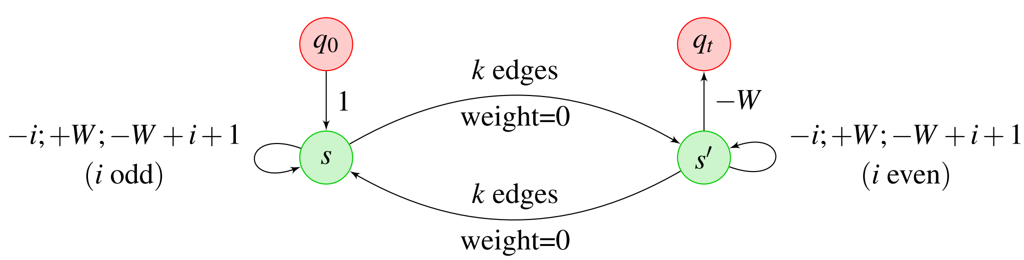

Figure 6: An example showing that more than one cycle per state can be needed.

Let us look at the universal cycles that we have in this arena: besides the cycles made of the $2k$ edges with weight zero between s and s', the only possible universal cycles have to depart from the first state of each cycle c_i (as they are the only states having a positive outgoing edge). Following Lemma 15, such cycles can be iterated arbitrarily many times, and set the energy level to some value in $[L;W]$. Since the only edge available at the end of a universal cycle has weight $+W$, the exact value of the universal cycles is irrelevant: the energy level will be W anyway when reaching the second state of each cycle c_i. As a consequence, using set*-edges in this example does not shorten the witnessing run, which then cannot be shorter than $3W+Wk+2$.*

We now move to the two-player setting. We begin with proving a result similar to Lemma 10:

Lemma 24 *Let G be a two-player arena, equipped with an LW-energy-reachability objective. Let q be a state of G, and $u \leq u'$ in $[L;W]$. If Player 1 wins the game from (q,u), then she also wins from (q,u').*

By Martin's theorem [24], our games are determined. It follows that if Player 2 wins from some configuration (q,v), she also wins from (q,v') for all $L \leq v' \leq v$ (assuming the contrary, i.e. (q,v') winning for Player 1, would lead to the contradictory statement that (q,v) is both winning for Player 1 and Player 2). Using classical techniques [9], we prove that Player 2 can be restricted to play memoryless strategies:

Proposition 25 *For two-player LW-energy-reachability games, memoryless strategies are sufficient for Player 2.*

A direct consequence of this result and of Theorem 22 is the following:

Corollary 26 *Two-player LW-energy-reachability games are in* coNP.

5 Energy reachability games with soft upper bound

We now consider games with limited violations, i.e. (reachability) games with $\mathsf{LSU}^{\#}$-energy, $\mathsf{LSU}^{\bar{\#}}$-energy and LSU^{Σ}-energy objectives. We address the problems of deciding the winner in the one-player and two-player settings, and consider the existence and minimization questions.

Theorem 27 *$\mathsf{LSU}^{\#}$-energy, $\mathsf{LSU}^{\bar{\#}}$-energy and LSU^{Σ}-energy (reachability) games are* PSPACE*-complete for one-player arenas, and* EXPTIME*-complete for two-player arenas.*

Proof. Membership in PSPACE and EXPTIME can be obtained by building a variant G_{LSU} of the G_{LU} arena: besides storing the energy level in each state, we can also store the amount of violations (for any of the three measures we consider). More precisely, given an arena G, lower and upper bounds L and U on the energy level, a soft bound S, and a bound V on the measure of violations, we define a new arena[3] G_{LSU} with set of states $(Q \times ([L;U] \cup \{\bot\}) \times ([0;V] \cup \{\bot\})^3)$, and each transition (q,w,q') of the original arena generates a transition from state $(q,l,(n,c,s))$ to state $(q',l',(n',c',s'))$ whenever

- l' correctly encodes the evolution of the energy level: if l and $l+w$ are in $[L;U]$, then $l'=l+w$; if the energy level leaves interval $[L;U]$ ($l+w<L$ or $l+w>U$) or has formerly leaved interval $[L;U]$ (in this case $l=\bot$), then $l'=\bot$.
- n' correctly stores the number of violations: $n'=\bot$ if $l' \in (S;U]$ and $n+1>V$ (the number of violations allowed is exceeded); once the number of violations is exceeded ($n=\bot$) or the maximal energy level is exceeded ($l'=\bot$), we have $n'=\bot$; last, $n'=n$ if $l' \in [L;S]$ (the current state does not violate bound S), and $n'=n+1$ if $l' \in (S;U]$ and $n+1 \leq V$ (the current state is an additional violation of soft bound S);

[3] In order to factor our proof, we store all three measures of violations in one single arena, even if only one measure per type of LSU-energy game is needed.

We can similarly update c' to count the current number of consecutive violations, and s' for the sum of all violations. In the resulting arena, values n, c and s keep tack of the number of violations, number of consecutive violations and sum of violations; their values range in $[0,V]$ and are set to $\perp$ as soon as they exceed bound V, or if the energy level has exceeded its bounds. The arena G_{LSU} has size exponential, and $\mathsf{LSU}^\star$-energy-reachability problems can then be reduced to solving reachability of corresponding sets of states in G_{LSU}. The announced complexity results follow. Hardness results are obtained by a straightforward encoding of LU-energy reachability problems, taking $S = U$ and $V = 0$.

Solving $\mathsf{LSU}^\#$-energy, $\mathsf{LSU}^{\overline{\#}}$-energy, LSU^Σ-energy games (without reachability objective) can be performed with arena G_{LSU} built above. Now, the objective in $\mathsf{LSU}^\#$-energy, $\mathsf{LSU}^{\overline{\#}}$-energy, LSU^Σ-energy games is to enforce infinite runs, that avoid states with $l = \perp$ and with $n = \perp$, $c = \perp$ or $s = \perp$, depending on the chosen measure of violations. □

When the strong upper bound U is not given, the existence problem consists in deciding if such a bound exists under which Player 1 wins the $\mathsf{LSU}^\star$-energy game. We have:

Theorem 28 *The existence problems for $\mathsf{LSU}^\#$-energy, $\mathsf{LSU}^{\overline{\#}}$-energy, and LSU^Σ-energy (reachability) games are* PSPACE *complete for the one-player case and* EXPTIME*-complete for the two-player case.*

Proof. Along any outcome of a winning strategy, the energy level remains below $S + V \cdot w_{\max}$, where $w_{\max}$ is the maximal weight appearing on transitions of G. This gives a strong upper bound U, with which we can apply the construction above and check the existence of a winning strategy for Player 1. □

Theorem 29 *Let G be an arena, L and S be integer bounds, and $V_{\max}$ be an integer. There exist algorithms that compute the value of U (if any) that minimizes the value of V (below $V_{\max}$) for which Player 1 has a winning strategy in a $\mathsf{LSU}^\star$-energy (reachability) game. These algorithms run in* PSPACE *for one-player games and in* EXPTIME *for two-player games. These bounds are sharp.*

Proof. We perform a binary search for an optimal value for V when the strict upper energy bound U varies between S and $S + V_{\max}.w_{\max}$. For each value U, we discard from G_{LSU} transitions for which the energy level exceeds U. One can remark that when U grows, the minimal amount of violation may decrease, both in reachability and infinite-run games. We can hence discover optimal values with a polynomial number of PSPACE checks (for the one-player games), and EXPTIME checks in the two-player case. □

6 Conclusion

This paper has considered several variants of energy games. The first variant defines games with upper and lower bound constraints, combined with reachability objectives. The second variant defines games with a strong lower bound and a soft upper bound, which can be temporarily exceeded. In the one player case, complexities ranges from PTIME to PSPACE-complete, and in the two-player case from $\mathsf{NP} \cap co\mathsf{NP}$ to EXPTIME-complete. In general, the complexity is the same for a reachability and for an infinite run objective. Interestingly, for LW-energy games, the complexity of the single player case is PTIME, but reachability objectives require exponential memory (in the size of the weak upper bound) while strategies are memoryless for infinite run objectives.

A possible extension of this work is to consider energy games with mean-payoff functions and discounted total payoff, both for the energy level and for the violation constraints, and the associated minimization and existence problems.

Acknowledgements. This work was supported by UMI Relax. We thank the anonymous reviewers for their suggestions, which helped us improve the presentation.

References

[1] Daniel Andersson (2006): *An improved algorithm for discounted payoff games*. In Janneke Huitink & Sophia Katrenko, editors: *Proceedings of the 11th ESSLLI Student Session*, pp. 91–98.

[2] Patricia Bouyer, Uli Fahrenberg, Kim Guldstrand Larsen & Nicolas Markey (2010): *Timed Automata with Observers under Energy Constraints*. In Karl Henrik Johansson & Wang Yi, editors: *Proceedings of the 13th International Workshop on Hybrid Systems: Computation and Control (HSCC'10)*, ACM Press, pp. 61–70, doi:10.1145/1755952.1755963.

[3] Patricia Bouyer, Uli Fahrenberg, Kim Guldstrand Larsen, Nicolas Markey & Jiří Srba (2008): *Infinite Runs in Weighted Timed Automata with Energy Constraints*. In Franck Cassez & Claude Jard, editors: *Proceedings of the 6th International Conferences on Formal Modelling and Analysis of Timed Systems (FORMATS'08)*, *Lecture Notes in Computer Science* 5215, Springer-Verlag, pp. 33–47, doi:10.1007/978-3-540-85778-5_4.

[4] Patricia Bouyer, Piotr Hofman, Nicolas Markey, Mickael Randour & Martin Zimmermann (2017): *Bounding Average-energy Games*. In Javier Esparza & Andrzej Murawski, editors: *Proceedings of the 20th International Conference on Foundations of Software Science and Computation Structure (FoSSaCS'17)*, *Lecture Notes in Computer Science* 10203, Springer-Verlag, pp. 179–195, doi:10.1007/978-3-662-54458-7_11.

[5] Patricia Bouyer, Kim Guldstrand Larsen & Nicolas Markey (2012): *Lower-Bound Constrained Runs in Weighted Timed Automata*. In: *Proceedings of the 9th International Conference on Quantitative Evaluation of Systems (QEST'12)*, IEEE Comp. Soc. Press, pp. 128–137, doi:10.1109/QEST.2012.28.

[6] Patricia Bouyer, Nicolas Markey, Mickael Randour, Kim Guldstrand Larsen & Simon Laursen (2015): *Average-energy games*. In Javier Esparza & Enrico Tronci, editors: *Proceedings of the 6th International Symposium on Games, Automata, Logics and Formal Verification (GandALF'15)*, *Electronic Proceedings in Theoretical Computer Science* 193, pp. 1–15, doi:10.4204/EPTCS.193.1.

[7] Franck Cassez, Jan J. Jensen, Kim Guldstrand Larsen, Jean-François Raskin & Pierre-Alain Reynier (2009): *Automatic Synthesis of Robust and Optimal Controllers – An Industrial Case Study*. In Rupak Majumdar & Paulo Tabuada, editors: *Proceedings of the 12th International Workshop on Hybrid Systems: Computation and Control (HSCC'09)*, *Lecture Notes in Computer Science* 5469, Springer-Verlag, pp. 90–104, doi:10.1007/978-3-642-00602-9_7.

[8] Arindam Chakrabarti, Luca de Alfaro, Thomas A. Henzinger & Mariëlle Stoelinga (2003): *Resource Interfaces*. In Rajeev Alur & Insup Lee, editors: *Proceedings of the 3rd International Conference on Embedded Software (EMSOFT'03)*, *Lecture Notes in Computer Science* 2855, Springer-Verlag, pp. 117–133, doi:10.1007/978-3-540-45212-6_9.

[9] Krishnendu Chatterjee & Laurent Doyen (2012): *Energy Parity Games*. *Theoretical Computer Science* 458, pp. 49–60, doi:10.1016/j.tcs.2012.07.038.

[10] Krishnendu Chatterjee, Laurent Doyen & Thomas A. Henzinger (2017): *The Cost of Exactness in Quantitative Reachability*. In Luca Aceto, Giorgio Bacci, Giovanni Bacci, Anna Ingólfsdóttir, Axel Legay & Radu Mardare, editors: *Models, Algorithms, Logics and Tools: Essays Dedicated to Kim Guldstrand Larsen on the Occasion of His 60th Birthday*, *Lecture Notes in Computer Science* 10460, Springer-Verlag, pp. 367–381, doi:10.1007/978-3-319-63121-9_18.

[11] Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger & Jean-François Raskin (2010): *Generalized Mean-payoff and Energy Games*. In Kamal Lodaya & Meena Mahajan, editors: *Proceedings of the 30th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'10)*, *Leibniz International Proceedings in Informatics* 8, Leibniz-Zentrum für Informatik, pp. 505–516, doi:10.4230/LIPIcs.FSTTCS.2010.505.

[12] Krishnendu Chatterjee, Mickael Randour & Jean-François Raskin (2014): *Strategy Synthesis for Multi-dimensional Quantitative Objectives*. *Acta Informatica* 51(3-4), pp. 129–163, doi:10.1007/s00236-013-0182-6.

[13] Aldric Degorre, Laurent Doyen, Raffaella Gentilini, Jean-François Raskin & Szymon Toruńczyk (2010): *Energy and Mean-Payoff Games with Imperfect Information*. In Anuj Dawar & Helmut Veith, editors:

Proceedings of the 24th International Workshop on Computer Science Logic (CSL'10), *Lecture Notes in Computer Science* 6247, Springer-Verlag, pp. 260–274, doi:10.1007/978-3-642-15205-4_22.

[14] Dario Della Monica & Aniello Murano (2018): *Parity-energy ATL for Qualitative and Quantitative Reasoning in MAS*. In Elisabeth André, Sven Koenig, Mehdi Dastani & Gita Sukthankar, editors: *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'18)*, International Foundation for Autonomous Agents and Multiagent Systems, pp. 1441–1449.

[15] Andrzej Ehrenfeucht & Jan Mycielski (1979): *Positional strategies for mean payoff games*. *International Journal of Game Theory* 8(2), pp. 109–113, doi:10.1007/BF01768705.

[16] Daniel Ejsing-Dunn & Lisa Fontani (2013): *Infinite Runs in Recharge Automata*. Master's thesis, Computer Science Department, Aalborg University, Denmark.

[17] Uli Fahrenberg, Line Juhl, Kim Guldstrand Larsen & Jiří Srba (2011): *Energy Games in Multiweighted Automata*. In Antonio Cerone & Pekka Pihlajasaari, editors: *Proceedings of the 8th International Colloquium on Theoretical Aspects of Computing (ICTAC'11)*, *Lecture Notes in Computer Science* 6916, Springer-Verlag, pp. 95–115, doi:10.1007/978-3-642-23283-1_9.

[18] John Fearnley & Marcin Jurdziński (2013): *Reachability in two-clock timed automata is PSPACE-complete*. In Fedor V. Fomin, Rusins Freivalds, Marta Kwiatkowska & David Peleg, editors: *Proceedings of the 40th International Colloquium on Automata, Languages and Programming (ICALP'13) – Part II*, *Lecture Notes in Computer Science* 7966, Springer-Verlag, pp. 212–223, doi:10.1007/978-3-642-39212-2_21.

[19] Stefan Göller, Christoph Haase, Joël Ouaknine & James Worrell (2010): *Model Checking Succinct and Parametric One-Counter Automata*. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide & Paul G. Spirakis, editors: *Proceedings of the 37th International Colloquium on Automata, Languages and Programming (ICALP'10) – Part II*, *Lecture Notes in Computer Science* 6199, Springer-Verlag, pp. 575–586, doi:10.1007/978-3-642-14162-1_48.

[20] Paul Hunter (2015): *Reachability in Succinct One-Counter Games*. In Mikołaj Bojańczyk, Sławomir Lasota & Igor Potapov, editors: *Proceedings of the 9th Workshop on Reachability Problems in Computational Models (RP'15)*, *Lecture Notes in Computer Science* 9328, Springer-Verlag, pp. 37–49, doi:10.1007/978-3-319-24537-9_5.

[21] Line Juhl, Kim Guldstrand Larsen & Jean-François Raskin (2013): *Optimal Bounds for Multiweighted and Parametrised Energy Games*. In Zhiming Liu, Jim Woodcock & Yunshan Zhu, editors: *Theories of Programming and Formal Methods – Essays Dedicated to Jifeng He on the Occasion of His 70th Birthday*, *Lecture Notes in Computer Science* 8051, Springer-Verlag, pp. 244–255, doi:10.1007/978-3-642-39698-4_15.

[22] Marcin Jurdziński, François Laroussinie & Jeremy Sproston (2007): *Model Checking Probabilistic Timed Automata with One or Two Clocks*. In Orna Grumberg & Michael Huth, editors: *Proceedings of the 13th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'07)*, *Lecture Notes in Computer Science* 4424, Springer-Verlag, pp. 170–184, doi:10.1007/978-3-540-71209-1_15.

[23] Marcin Jurdziński, Ranko Lazić & Sylvain Schmitz (2015): *Fixed-Dimensional Energy Games are in Pseudo-Polynomial Time*. In Magnús M. Halldórsson, Kazuo Iwana, Naoki Kobayashi & Bettina Speckmann, editors: *Proceedings of the 42nd International Colloquium on Automata, Languages and Programming (ICALP'15) – Part II*, *Lecture Notes in Computer Science* 9135, Springer-Verlag, pp. 260–272, doi:10.1007/978-3-662-47666-6_21.

[24] Donald A. Martin (1975): *Borel Determinacy*. *Annals of Mathematics* 102(2), pp. 363–371, doi:10.2307/1971035.

[25] Julien Reichert (2016): *On The Complexity of Counter Reachability Games*. *Fundamenta Informaticae* 143(3-4), pp. 415–436, doi:10.3233/FI-2016-1320.

[26] Yaron Velner, Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, Alexander Rabinovich & Jean-François Raskin (2015): *The complexity of multi-mean-payoff and multi-energy games*. *Information and Computation* 241, pp. 177–196, doi:10.1016/j.ic.2015.03.001.

[27] Uri Zwick & Mike Paterson (1996): *The Complexity of Mean Payoff Games on Graphs*. Theoretical Computer Science 158(1-2), pp. 343–359, doi:10.1016/0304-3975(95)00188-3.