Adding Negative Prices to Priced Timed Games*

Thomas Brihaye[1], Gilles Geeraerts[2,**], Shankara Narayanan Krishna[3], Lakshmi Manasa[3], Benjamin Monmege[2], and Ashutosh Trivedi[3]

[1] Université de Mons, Belgium
thomas.brihaye@umons.ac.be
[2] Université libre de Bruxelles, Belgium
{gigeerae,benjamin.monmege}@ulb.ac.be
[3] IIT Bombay, India
{krishnas,manasa,trivedi}@cse.iitb.ac.in

Abstract. Priced timed games (PTGs) are two-player zero-sum games played on the infinite graph of configurations of priced timed automata where two players take turns to choose transitions in order to optimize cost to reach target states. Bouyer et al. and Alur, Bernadsky, and Madhusudan independently proposed algorithms to solve PTGs with nonnegative prices under certain divergence restriction over prices. Brihaye, Bruyère, and Raskin later provided a justification for such a restriction by showing the undecidability of the optimal strategy synthesis problem in the absence of this divergence restriction. This problem for PTGs with one clock has long been conjectured to be in polynomial time, however the current best known algorithm, by Hansen, Ibsen-Jensen, and Miltersen, is exponential. We extend this picture by studying PTGs with both negative and positive prices. We refine the undecidability results for optimal strategy synthesis problem, and show undecidability for several variants of optimal reachability cost objectives including reachability cost, time-bounded reachability cost, and repeated reachability cost objectives. We also identify a subclass with bi-valued price-rates and give a pseudo-polynomial (polynomial when prices are nonnegative) algorithm to partially answer the conjecture on the complexity of one-clock PTGs.

1 Introduction

Timed automata [2] equip finite automata with a finite number of real-valued variables—aptly called clocks—that evolve with a uniform rate. The syntax of timed automata also permits specifying *transition guards* and *location (state) invariants* using the constraints over clock valuations, and resetting the clocks as a means to remember the time since the execution of a transition. Timed automata is a well-established formalism to specify time-critical properties of real-time systems. Priced timed automata [3,4] (PTAs) extend timed automata with

* The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under Grant Agreement n601148 (CASSTING).
** Supported by a 'Crédit aux chercheurs' number 1808881 of the F.R.S./FNRS.

P. Baldan and D. Gorla (Eds.): CONCUR 2014, LNCS 8704, pp. 560–575, 2014.

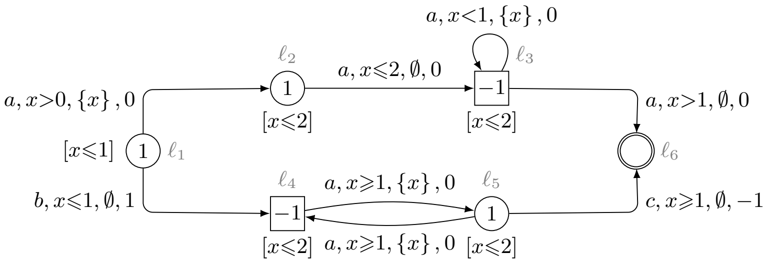

Fig. 1. A price timed game arena with one clock

price information by augmenting locations with price-rates and transitions with discrete prices. The natural reachability-cost optimization problem for PTAs is known to be decidable with the same complexity [6] as the reachability problem (PSPACE-complete), and forms the backbone of many applications of timed automata including scheduling and planning.

Priced timed games (PTGs) extend the reachability-cost optimization problem to the setting of competitive optimization problem, and form the basis of optimal controller synthesis [19] for real-time systems. We study turn-based variant of these games where the game arena is a PTA with a partition of the locations between two players Player 1 and Player 2. A play of such a game begins with a token in an initial location, and at every step the player controlling the current location proposes a valid timed move, i.e., a time delay and a discrete transition, and the state of the system is modified accordingly. The play stops if the token reaches a location from a distinguished set of *target locations*, and the payoff of the play is equal to the cost accumulated before reaching the target location. If the token never reaches a target location then the game continues forever, and the payoff in this case is $+\infty$ irrespective of actual cost of the infinite play. We characterize a PTG according to the objectives of Player 1. Since we study zero-sum games, the objective of Player 2 is also implicitly defined. We study PTGs with the following objectives: (i) *Constrained-price reachability* objective $\mathsf{Reach}(\bowtie K)$ is to achieve a payoff C of the play such that $C \bowtie K$ where $\bowtie \in \{\leqslant, <, =, >, \geqslant\}$ and $K \in \mathbb{N}$; (ii) *Bounded-time reachability* objective $\mathsf{TBReach}(K, T)$ is to keep the payoff of the play less than K while keeping the total time elapsed within T units; and (iii) *Repeated reachability* objective $\mathsf{RReach}(\eta)$ is to visit target infinitely often with a payoff in the interval $[-\eta, \eta]$.

An example of PTG with clock variable x and six locations is given in Fig. 1. We depict Player 1 locations as circles and Player 2 locations as boxes. The numbers inside locations denote their price-rates, while the clock constraints next to a location depicts its invariant. We denote a transition, as usual, by an arrow between two location annotated by a tuple a, g, r, c where a is the label, g is the guard, r is the clocks reset set, and c is the cost of the transition.

Related Work. PTGs with constrained-price reachability objective $\mathsf{Reach}(\leqslant K)$ were independently introduced in [9] and [1], with semi-algorithms to decide the existence of winning strategy for Player 1 in PTGs with nonnegative prices.

They also showed that under the *strongly non-Zeno assumption* on prices the proposed semi-algorithms always terminate. This assumption was justified in [11] by showing that, in the absence of non-Zeno assumption, the problem of deciding the existence of winning strategy for the objective $\mathsf{Reach}(\leqslant K)$ is undecidable for PTGs with five or more clocks. This result has been later refined in [7] by showing that the problem is undecidable for PTGs with three or more clocks and nonnegative prices. In [5] is showed the undecidability of the existence of winning strategy problem for $\mathsf{Reach}(\leqslant K)$ objective over PTGs with both positive and negative price-rates and two or more clocks.

On a positive side, the existence of winning strategy for $\mathsf{Reach}(\leqslant K)$ problem for PTGs with one clock when the price-rates are restricted to values 0 and $d \in \mathbb{N}$ has been shown decidable in [11], by proving that the semi-algorithms in [9,1] always terminate. However, the authors did not provide any complexity analysis of their algorithm. One-clock PTGs with nonnegative prices are reconsidered in [10], and a 3-EXPTIME algorithm is given to solve the problem, while the best known lower bound is PTIME. A tighter analysis of the problem is presented in [20] that lowered the known complexity of this problem to EXPTIME, namely $2^{O(n^2+m)}$ where n is the number of locations and m is the number of transitions. A significant improvement over the complexity $(m12^n n^{O(1)})$ was given in [15] by improving the analysis of the semi-algorithms by [9,1].

Contributions. We consider PTGs with both negative and positive prices. We show that deciding the existence of a winning strategy for reachability objective $\mathsf{Reach}(\bowtie K)$ is undecidable for PTGs with two or more clocks. In [18], a theory of time-bounded verification has been proposed, arguing that restriction to bounded-time domain reclaims the decidability of several key verification problems. As an example, we cite [12] where authors recovered the decidability of the reachability problem for hybrid automata under time-bounded restriction. We begin studying PTGs with bounded reachability objective $\mathsf{TBReach}(K,T)$ hoping that the problem may be decidable due to time-bounded restriction. However, we answer this question negatively by showing undecidability of the existence of winning strategy problem for PTGs with six or more clocks. We also show the undecidability for the corresponding problem for repeated reachability objective $\mathsf{RReach}(\eta)$ for PTGs with three or more clocks.

On the positive side, we introduce a previously unexplored subclass of one-clock PTGs, called one-clock bi-valued priced timed games (1BPTGs), where the price-rates of locations are taken from a set of two integers from $\{-d, 0, d\}$ (with d any positive integer). None of the previously cited algorithms can be applied in this case since we do not assume non-Zenoness of prices and consider both positive and negative prices. After showing a determinacy result for 1BPTGs, we proceed to give a pseudo-polynomial time algorithm to compute the value and ε-optimal strategy for both players with $\mathsf{Reach}(\leqslant K)$ objective. The complexity drops to polynomial for 1BPTGs if the price-rates are non-negative integers. This gives a polynomial time algorithm for the one-clock PTG problem studied in [11]. Due to lack of space, full proofs of the results are given in [13].

2 Reachability-Cost Games on Priced Game Graphs

PTGs can be considered as a succinct representation of some games on uncountable state space characterized by the configuration graph of timed automata.

We begin by introducing the concepts and notations related to such more general game arenas that we call priced game graphs.

Definition 1. *A* priced game graph *is a tuple* $\mathcal{G} = (V, A, E, \pi, V_f)$ *where:*
- $V = V_1 \uplus V_2$ *is the set of vertices partitioned into the sets* V_1 *and* V_2*;*
- A *is a set of labels called actions;*
- $E \colon V \times A \to V$ *is the edge function defining the set of labeled edges;*
- $\pi \colon V \times A \to \mathbb{R}$ *is the price function that assigns prices to edges; and*
- $V_f \subseteq V$ *is the set of target vertices.*

We call a game graph finite *if both* V *and* A *are finite and with rational prices.*

A reachability-cost game begins with a token placed on some initial vertex v_0. At each round, the player who controls the current vertex v chooses an action $a \in A$ and the token is moved to the vertex $E(v, a)$. The two players continue moving the token in this fashion, and give rise to an infinite sequence of vertices and actions called a play of the game. Formally, a finite play r is a finite sequence of vertices and actions $\langle v_0, a_0, v_1, a_1, \ldots, a_{n-1}, v_n\rangle$ where for each $0 \leqslant i < n$ we have that $v_{i+1} = E(v_i, a_i)$; we write $\mathsf{Last}(r)$ for the last vertex of a finite play, here $\mathsf{Last}(r) = v_n$. An infinite play is defined analogously. We write $\mathsf{FPlay}_{\mathcal{G}}$ ($\mathsf{FPlay}_{\mathcal{G}}(v)$) for the set of finite plays (starting from the vertex v) of the game graph $\mathcal{G}$. We often omit the subscript when the game arena is clear form the context. We similarly define Play and $\mathsf{Play}(v)$ for the set of infinite plays. For all $k \geqslant 0$, we let $r[k]$ be the prefix $\langle v_0, a_0, \ldots, a_{k-1}, v_k\rangle$ of r, and we denote by $\mathsf{Cost}(r[k]) = \sum_{i=0}^{k-1} \pi(v_i, a_i)$ its cost. We write $\mathsf{Stop}(r)$ for the index of the first target vertex in r, i.e., $\mathsf{Stop}(r) = \inf\{k : v_k \in F\}$. We define the cost of an infinite run $r = \langle v_0, a_1, v_1, \ldots\rangle$ as $\mathsf{Cost}(r) = +\infty$ if $\mathsf{Stop}(r) = \infty$ and $\mathsf{Cost}(r) = \mathsf{Cost}(r[\mathsf{Stop}(r)])$, otherwise.

A strategy for a Player i (for $i \in \{1, 2\}$) is a partial function $\sigma : \mathsf{FPlay} \to A$ that is defined for a run $r = \langle v_0, a_0, v_1, \ldots, a_{n-1}, v_n\rangle$ if $v_n \in V_i$ and is such that $E(v_n, \sigma(r))$ is defined, i.e., there is a $\sigma(r)$-labeled outgoing transition from v_n. We denote by $\mathsf{Strat}_i(\mathcal{G})$ (or Strat_i when the game arena is clear) the set of strategies for Player i. Given a strategy profile $(\sigma_1, \sigma_2) \in \mathsf{Strat}_1 \times \mathsf{Strat}_2$ for both players, and an initial vertex $v \in V$, the unique infinite play $\mathsf{Play}(v, \sigma_1, \sigma_2) = \langle v_0, a_0, v_1, \ldots v_k, a_k, v_{k+1}, \ldots\rangle$ is such that for all $k \geqslant 0$ if $v_k \in V_i$, for $i = 1, 2$, then $a_{k+1} = \sigma_i(r[k])$ and $v_{k+1} = E(v_k, a_{k+1})$. A strategy σ is said to be *memoryless* (or *positional*) if, for all finite plays $r, r' \in \mathsf{FPlay}$ with $\mathsf{Last}(r) = \mathsf{Last}(r')$ we have that $\sigma(r) = \sigma(r')$. Similarly, *finite-memory strategies* can be defined as implementable with Moore machines, see [14] for a formal definition.

We consider optimal reachability-cost games on priced game graphs, where the goal of Player 1 is to minimize the reachability-cost, while the goal of Player 2 is the opposite. The standard concepts of upper value and lower value of the optimal reachability-cost game are defined in straightforward manner. Formally, the upper-value $\overline{\mathsf{Val}}_{\mathcal{G}}(v)$ and lower value $\underline{\mathsf{Val}}_{\mathcal{G}}(v)$ of a game starting from a vertex v is defined as $\overline{\mathsf{Val}}_{\mathcal{G}}(v) = \inf_{\sigma_1 \in \mathsf{Strat}_1} \sup_{\sigma_2 \in \mathsf{Strat}_2} \mathsf{Cost}(\mathsf{Play}(v, \sigma_1, \sigma_2))$ and $\underline{\mathsf{Val}}_{\mathcal{G}}(v) = \sup_{\sigma_2 \in \mathsf{Strat}_2} \inf_{\sigma_1 \in \mathsf{Strat}_1} \mathsf{Cost}(\mathsf{Play}(v, \sigma_1, \sigma_2))$. It is easy to see that $\underline{\mathsf{Val}}_{\mathcal{G}}(v) \leqslant \overline{\mathsf{Val}}_{\mathcal{G}}(v)$ for every vertex v. We say that a game is *determined* if the lower and the upper values match for every vertex v, and in this case, we say that the optimal value of the game exists and we let $\mathsf{Val}_{\mathcal{G}}(v) = \underline{\mathsf{Val}}_{\mathcal{G}}(v) = \overline{\mathsf{Val}}_{\mathcal{G}}(v)$. The determinacy of these games follow from Martin's determinacy theorem, and an alternative proof is given in [14].

In the following, we write $\mathsf{Cost}(v, \sigma_1)$ for the value of the strategy σ_1 of Player 1 from vertex v, i.e., $\mathsf{Cost}(v, \sigma_1) = \sup_{\sigma_2 \in \mathsf{Strat}_2} \mathsf{Cost}(\mathsf{Play}(v, \sigma_1, \sigma_2))$. A strategy σ_1^* of Player 1 is said to be optimal from v if $\mathsf{Cost}(v, \sigma_1^*) = \overline{\mathsf{Val}}_{\mathcal{G}}(v)$. Optimal strategies do not always exist, hence we also define ε-optimal strategies. For $\varepsilon > 0$, a strategy σ_1 is an ε-optimal strategy if for all vertex $v \in V$, $\mathsf{Cost}(v, \sigma_1) \leqslant \overline{\mathsf{Val}}_{\mathcal{G}}(v) + \varepsilon$. In this paper we exploit the following result from [14].

Theorem 1 ([14]). *Let $\mathcal{G}$ be a finite priced game graph.*
1. *Deciding $\mathsf{Val}_{\mathcal{G}}(v) = +\infty$ is in Polynomial Time.*
2. *Deciding $\mathsf{Val}_{\mathcal{G}}(v) = -\infty$ is in* NP $\cap$ co-NP*, can be achieved in pseudo-polynomial time*[1] *and is as hard as solving mean-payoff games [21].*
3. *Given $-\infty < \mathsf{Val}_{\mathcal{G}}(v) < +\infty$ for every vertex v, optimal strategies exist for both players. In particular, Player 2 has optimal memoryless strategies, while Player 1 has optimal finite-memory strategies. Moreover, the values $\mathsf{Val}_{\mathcal{G}}(v)$, as well as optimal strategies, can be computed in pseudo-polynomial time.*

It must be noticed that, in the presence of negative costs, and even when every vertex v has a finite value $\mathsf{Val}_{\mathcal{G}}(v) \in \mathbb{R}$, memoryless optimal strategies may not exist for Player 1, as pointed out in [14, Example 1].

3 Priced Timed Games

In order to formally introduce priced timed games, we need to define the concepts of clocks, clock valuations, constraints, and zones. Let $\mathcal{X}$ be a finite set of real-valued variables called *clocks*. A clock valuation on $\mathcal{X}$ is a function $\nu \colon \mathcal{X} \to \mathbb{R}_{\geqslant 0}$ and we write $V(\mathcal{X})$ for the set of clock valuations. Abusing notation, we also treat a valuation ν as a point in $\mathbb{R}^{|\mathcal{X}|}$. If $\nu \in V(\mathcal{X})$ and $t \in \mathbb{R}_{\geqslant 0}$ then we write $\nu + t$ for the clock valuation defined by $(\nu + t)(c) = \nu(c) + t$ for all $c \in \mathcal{X}$. For $C \subseteq \mathcal{X}$, we write $\nu[C := 0]$ for the valuation where $\nu[C := 0](c)$ equals 0 if $c \in C$ and $\nu(c)$ otherwise. A clock constraint over $\mathcal{X}$ is a conjunction of simple constraints of the form $c \bowtie i$ or $c - c' \bowtie i$, where $c, c' \in \mathcal{X}$, $i \in \mathbb{N}$ and $\bowtie \in \{<, >, =, \leqslant, \geqslant\}$. A clock zone is a finite set of clock constraints that defines a convex set of clock valuations. We write $Z(\mathcal{X})$ for the set of clock zones over the set of clocks $\mathcal{X}$.

Definition 2. *A priced timed game is a tuple $\mathcal{A} = (L, \mathcal{X}, \mathrm{Inv}, \Sigma, \delta, \omega, L_f)$ where:*
- *$L = L_1 \uplus L_2$ is a finite set of locations, partitioned into the sets L_1 and L_2;*
- *$\mathcal{X}$ is a finite set of clocks;*

[1] Polynomial time if the prices are encoded in unary.

– Inv: $L \to Z(\mathcal{X})$ *associates an invariant to each location;*
– Σ *is a finite set of labels;*
– $\delta\colon L \times \Sigma \to Z(\mathcal{X}) \times 2^{\mathcal{X}} \times L$ *is a transition function that maps a location $\ell \in L$ and label $a \in \Sigma$ to a clock zone $\zeta \in Z(\mathcal{X})$ representing the guard on the transition, a set of clocks $R \subseteq \mathcal{X}$ to be reset and successor location $\ell' \in L$;*
– $\omega\colon L \cup \Sigma \to \mathbb{Z}$ *is the price function; and*
– *and $L_f \subseteq L$ is the set of target locations.*

A configuration of a PTG is a tuple $(\ell, \nu) \in L \times V$ where ℓ is a location, ν is a clock valuation and $\nu \in \mathrm{Inv}(\ell)$. A timed action is a tuple $\tau = (t, a) \in \mathbb{R}_{\geqslant 0} \times \Sigma$ where t is a time delay and a is a label. In the following, for a timed move $\tau = (t, a) \in \mathbb{R}_{\geqslant 0} \times \Sigma$, we let $\mathsf{del}(\tau) = t$ be the delay part and $\mathsf{lab}(\tau) = a$ be the label part. The semantics of a PTG is given as an infinite priced game graph.

Definition 3 (Semantics). *The semantics of a PTG $\mathcal{A} = (L, \mathcal{X}, \mathrm{Inv}, \Sigma, \delta, \omega, L_f)$ is given as a priced game graph $[\![\mathcal{A}]\!] = (S, \Gamma, \Delta, \kappa, S_f)$ where*
– $S = \{(\ell, \nu) \in L \times V \mid \nu \in \mathrm{Inv}(\ell)\}$ *is the set of configurations of the PTG;*
– $\Gamma = \mathbb{R}_{\geqslant 0} \times \Sigma$ *is the set of timed moves;*
– $\Delta\colon S \times \Gamma \to S$ *is the transition function defined by $(\ell', \nu') = \Delta((\ell, \nu), (t, a))$ if $\delta(\ell, a) = (\zeta, R, \ell')$ such that $\nu + t \in \zeta$, $\nu + t' \in \mathrm{Inv}(\ell)$ for all $0 \leqslant t' \leqslant t$, and $\nu' = (\nu + t)[R := 0]$;*
– $\kappa : S \times \Gamma \to \mathbb{R}$ *is such that $\kappa((\ell, \nu), (t, a)) = \omega(\ell) \times t + \omega(a)$; and*
– $S_f \subseteq S$ *is such that $(\ell, \nu) \in S_f$ iff $\ell \in L_f$.*

The concepts of a play, its cost, and strategies of players for a PTG $\mathcal{A}$ is defined via corresponding objects for its semantic priced game graph $[\![\mathcal{A}]\!]$. In the previous section we introduced games with reachability-cost objective for priced game graphs. We also study the following winning objectives for Player 1 in the context of priced timed games; the objective for Player 2 is the opposite.

1. **Constrained-price reachability.** The constrained-price reachability objective $\mathsf{Reach}(\leqslant K)$ is to keep the payoff within a given bound $K \in \mathbb{N}$. Objectives $\mathsf{Reach}(\bowtie K)$ for constrains $\bowtie \in \{<, =, >, \geqslant\}$ are defined analogously.
2. **Bounded-time reachability.** Given constants $K, T \in \mathbb{N}$, the bounded-time reachability objective $\mathsf{TBReach}(K, T)$ is to keep the payoff of the play less than or equal to K while keeping the total time elapsed within T units.
3. **Repeated reachability.** For this objective, we consider slightly different semantics of the game where the play continues forever, and the repeated reachability objective $\mathsf{RReach}(\eta)$, $\eta \in \mathbb{R}_{\geqslant 0}$ is to visit target locations infinitely often each time with a payoff in a given interval $[-\eta, \eta]$.

In Section 4, we sketch the proof of the following negative result regarding the decidability of PTGs with these objectives. This result is particularly surprising for bounded-time reachability objective, since bounded-time restriction has been shown to recover decidability in many related problems [18,12].

Theorem 2. *Let $\mathcal{A}$ be a priced timed game arena. The decision problems corresponding to the existence of winning strategy for following objectives are undecidable:*

1. $\mathsf{Reach}(\bowtie K)$ *objective for PTGs with two or more clocks and arbitrary prices;*
2. $\mathsf{TBReach}(K,T)$ *objective for PTGs with five or more clocks; and prices 0,1;*
3. $\mathsf{RReach}(\eta)$ *objective for PTGs with three or more clocks and arbitrary prices.*

To recover decidability, we consider a subclass of one-clock PTGs. In this subclass, the set of clocks $\mathcal{X}$ is a singleton $\{x\}$, and price-rates of the locations come from a doubleton set $\{p^-, p^+\}$ with $p^- < p^+$ two distinct elements of $\{-1, 0, 1\}$ (no condition is made on the prices $\omega(a)$ of labels $a \in \Sigma$). We call these restricted games *one-clock bi-valued priced timed games*, abbreviated as 1PTG(p^-, p^+), or 1BPTG if p^- and p^+ do not matter. All our results may easily be extended to the case where p^- and p^+ are taken from the set $\{-d, 0, d\}$ with $d \in \mathbb{N}$. We devote Section 5 to the proof of the following decidability results.

Theorem 3. *We have the following results:*
1. 1BPTG*s are determined.*
2. *The value of a* 1BPTG *can be computed in pseudo-polynomial time.*
3. *Given that a* 1BPTG *has a finite value, an ε-optimal strategy for Player 1 can be computed in pseudo-polynomial time.*
4. *Aforementioned complexities drop to polynomial time for* 1PTG$(0,1)$ *with prices of labels taken from* $\mathbb{N}$.

4 Undecidability Results

In this section we provide a proof sketch of our undecidability result (Theorem 2) by reducing the halting problem for two counter machines (see [17]) to the existence of a winning strategy for Player 1 for the desired objective. For all the three objectives, given a two counter machine, we construct a PTG $\mathcal{A}$ whose building blocks are the modules for instructions. In these reductions the objective of Player 1 is linked to a faithful simulation of various increment, decrement, and zero-test instructions of the machine by choosing appropriate delays to adjust the clocks to reflect changes in counter values. The goal of Player 2 is then to verify the simulation performed by Player 1. Proofs of correctness of the reductions, as well as more details can be found in the appendix.

Constrained-Price Reachability Objectives $\mathsf{Reach}(\bowtie K)$. The result in the case $\mathsf{Reach}(\leqslant K)$ is a consequence of the result in [5]. Undecidability for other comparison operators $\bowtie$ is a new contribution. We only consider the objective $\mathsf{Reach}(=1)$ in this section, since proofs for other constraints are similar. Our reduction uses a PTG with two clocks x_1 and x_2, arbitrary price-rates for locations and no prices for labels. Each counter machine instruction (increment, decrement, and test for zero value) is specified using a PTG module. The main invariant in our reduction is that upon entry into a module, we have that $x_1 = \frac{1}{5^{c_1}7^{c_2}}$ and $x_2 = 0$ where c_1 (respectively, c_2) is the value of counter C_1 (respectively, C_2). We outline the simulation of a decrement instruction for counter C_1 in Fig. 2. Let us denote by $x_{old} = \frac{1}{5^{c_1}7^{c_2}}$ the value of x_1 while entering the

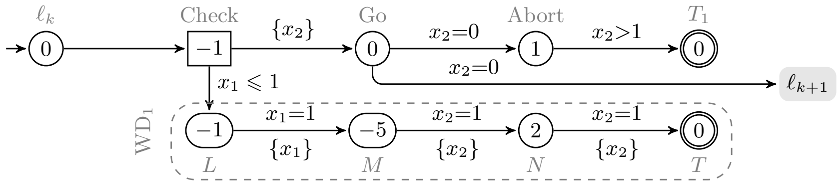

Fig. 2. Decrement module for the objection $\mathsf{Reach}(=1)$

module. At the location ℓ_{k+1} of the module, $x_1 = x_{new}$ should be $5x_{old}$ to correctly decrement counter C_1. At location ℓ_k, Player 1 spends a non-deterministic amount of time $t_k = x_{new} - x_{old}$ such that $x_{new} = 5x_{old} + \varepsilon$. To correctly decrement C_1, ε should be 0, and t_k must be $\frac{4}{5^{c_1}7^{c_2}}$. At location Check, Player 2 could choose to go to Go (in order to continue the simulation of the machine) or go to the widget WD$_1$, if he suspects that $\varepsilon \neq 0$. If Player 2 spends time $t > 0$ in the location Check before proceeding to Go, then Player 1 can enter the location Abort (to abort the simulation), rather than going to ℓ_{k+1}. Player 1 spends $1+t$ time in location Abort and reaches a target T_1 with cost 1 (and thus achieve his objective). However, if $t = 0$ then entering location Abort will make the cost to be greater than 1 (which is losing for Player 1). If Player 2 decides to enter widget WD$_1$, then the cost upon reaching the target in the widget WD$_1$ is $1+\varepsilon$ which is 1 iff $\varepsilon = 0$.

Bounded-Time Reachability Objective. We sketch the reduction for objective $\mathsf{TBReach}(K,T)$. Our reduction uses a PTG with price-rates 0 or 1 on locations, and zero prices on labels, along with five clocks x_1, x_2, z, a, b. On entry into a module for the $(k+1)$th instruction, we always have one of the two clocks x_1, x_2 with value $\frac{1}{2^{k+c_1}3^{k+c_2}}$ and other is 0. Clock z keeps track of the total time elapsed during simulation of an instruction: we always have $z = 1 - \frac{1}{2^k}$ at the end of simulating kth instruction. Thus, time $\frac{1}{2}$ is spent simulating the first instruction, $\frac{1}{4}$ for the second instruction and so on, so that the total time spent in simulating the main modules is less than 1. The main challenge here is to ensure that only a bounded time is spent along the entire simulation, along with updating the counter values correctly. Clocks a, b are used for rough work. For instance, if the $(k+1)$th instruction ℓ_{k+1} is an increment of C_1, and we have $x_1 = \frac{1}{2^{k+c_1}3^{k+c_2}}$, while $a = b = x_2 = 0$, and $z = 1 - \frac{1}{2^k}$, then at the end of the module simulating ℓ_{k+1}, we want $x_2 = \frac{1}{2^{k+1+c_1+1}3^{k+1+c_2}}$ and $x_1 = 0$ and $z = 1 - \frac{1}{2^{k+1}}$.

Repeated Reachability Objective. Finally, we consider the repeated reachability objective $\mathsf{RReach}(\eta)$. Our reduction uses a PTG with 3 clocks, and arbitrary price-rates, but zero prices for labels. On entry into a module, we have $x_1 = \frac{1}{5^{c_1}7^{c_2}}$, $x_2 = 0$ and $x_3 = 0$, where c_1, c_2 are the values of C_1 and C_2. Fig. 3 shows module to simulate decrement C_1. Location ℓ_k is entered with $x_1 = \frac{1}{5^{c_1}7^{c_2}}, x_2 = 0$ and $x_3 = 0$. To correctly decrement C_1, Player 1 should choose a delay of $\frac{4}{5^{c_1}7^{c_2}}$ in location ℓ_k. At location Check, no time can elapse

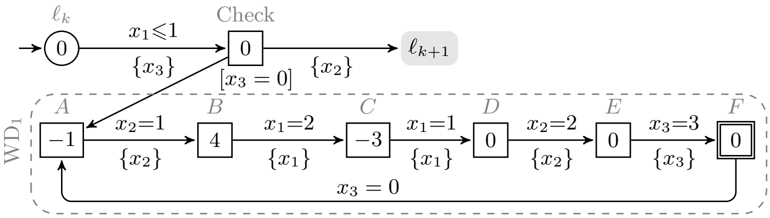

Fig. 3. Decrement module for Repeated reachability objective

because of the invariant. If Player 1 makes an error, and delays $\frac{4}{5^{c_1}7^{c_2}} + \varepsilon$ at ℓ_k ($\varepsilon \neq 0$) then Player 2 can jump in widget WD$_1$. The cost of going from location A to F is ε; each time we come back to A, the clock values with which A was entered are restored. Clearly, if $\varepsilon \neq 0$, Player 2 can incur a cost that is not in $[-\eta, \eta]$ by taking the loop from A to F a large number of times.

5 One-Clock Bi-Valued Priced Timed Games

This section is devoted to the proof of Theorem 3. First of all, let us assume that all 1BPTGs $\mathcal{A}$ we consider are *bounded*, i.e., that there is a global invariant in every location, of the form $x \leqslant M_K$ (where M_K denotes the greatest constant appearing in the clock guards and invariants of $\mathcal{A}$). This restriction comes w.l.o.g since every 1BPTG arena can be made bounded with a polynomial algorithm.[2]

Our proof of Theorem 3 is based on an extension of the classical notion of regions in timed automata, in the spirit of the regions introduced to define the corner point abstraction [8]. Indeed, to take the price into account, ε-optimal strategies do not take uniform decisions on the classical regions. That is why we need to subdivide each classical region into three parts: two small parts around the corners of the region (that we will call *borders* in the following, considering our one-clock setting), and a big part in-between. We will show that considering only strategies that never jump into those big parts is sufficient (Lemma 1). Lemma 2, later, shows a stronger result that one can restrict attention to strategies that play closer and closer to the borders of the regions as time elapses. Finally, we combine these results to show that a finite abstraction of 1BPTGs is sufficient to compute the value as well as ε-optimal strategies (Lemma 3). This not only yields the desired result, but also provides us further insight into the shape of ε-optimal strategies for both players.

5.1 Reduction to η-Region-Uniform Strategies

Since we only consider one-clock PTGs, we need not consider the standard Alur-Dill regional equivalence relation. Instead, we consider special region equivalence relation characterized by the intervals with constants appearing in guards

[2] By introducing auxiliary states in order to reset the clock x at every time unit once its value goes beyond M_k. The polynomial complexity holds only for one-clock PTGs.

and invariants of $\mathcal{A}$ inspired by Laroussinie, Markey, and Schnoebelen construction [16]. Let $0=M_0<M_1<\cdots<M_K$ be the integers appearing in guards and invariants of $\mathcal{A}$. We say that two valuations $\nu, \nu' \in \mathbb{R}_{\geqslant 0}$ are region-equivalent (or lie in the same region), and we write $\nu \sim \nu'$, if for every $k \in \{0, \dots, K\}$, $\nu \leqslant M_k$ iff $\nu' \leqslant M_k$, and $\nu \geqslant M_k$ iff $\nu' \geqslant M_k$. We define the set of regions to be the set of equivalence classes of $\sim$. We extend the equivalence relation $\sim$ from valuations to configurations in a straightforward manner. We also generalize the regional equivalence relation to the plays. For two (finite or infinite) plays $r = \langle(\ell_0, \nu_0), (t_0, a_0), \dots\rangle$ and $r' = \langle(\ell'_0, \nu'_0), (t'_0, a'_0), \dots\rangle$ we say that $r \sim r'$ if the lengths of r and r' are equal, and they define sequences of regional equivalent states (i.e., $(\ell_i, \nu_i) \sim (\ell'_i, \nu'_i)$ for all $i \geqslant 0$) and follow equivalent timed actions (i.e., $a_i = a'_i$ and $\nu_i + t_i \sim \nu'_i + t'_i$ for all $i \geqslant 0$). We also consider a refinement of region equivalence relation that we call the η-region equivalence relation, and we write $\sim_\eta$, for a given $\eta \in (0, \frac{1}{3})$. Intuitively, $\nu \sim_\eta \nu'$ if both valuations are close or far from any borders of the regions, with respect to the distance η.

Definition 4 (η-regions). *For valuations $\nu, \nu' \in \mathbb{R}_{\geqslant 0}$ we say that $\nu \sim_\eta \nu'$ if $\nu \sim \nu'$ and for every $k \in \{0, \dots, K-1\}$, $|\nu - M_k| \leqslant \eta$ iff $|\nu' - M_k| \leqslant \eta$, and $\nu \geqslant M_K - \eta$ iff $\nu' \geqslant M_K - \eta$. We assume the natural order $\preceq$ over η-regions by their lower bounds. We call η-regions the equivalence classes of $\sim_\eta$. We also extend the relation $\sim_\eta$ to configurations and runs.*

For instance, if $M_1 = 2$ and $M_2 = 3$, the set of η-regions is given by $\{\{0\}, (0, \eta], (\eta, 2-\eta), [2-\eta, 2), \{2\}, (2, 2+\eta], (2+\eta, 3-\eta), [3-\eta, 3), \{3\}, (3, +\infty)\}$. We next introduce the strategies of a restricted shape with the properties that they depend only on the η-region abstraction of runs; their decision is uniform over each η-region; and they play η-close to the borders of the regions.

Definition 5 (η-region uniform strategies). *Let $\eta \in (0, \frac{1}{3})$ be a constant. A strategy $\sigma \in \mathsf{Strat}_1 \cup \mathsf{Strat}_2$ is said to be* η-region-uniform *if*

- *for all finite run $r \sim_\eta r'$ ending respectively in (ℓ, ν) and (ℓ, ν') (in particular $\nu \sim_\eta \nu'$) we have $\nu + \mathsf{del}(\sigma(r)) \sim_\eta \nu' + \mathsf{del}(\sigma(r'))$ and $\mathsf{lab}(\sigma(r)) = \mathsf{lab}(\sigma(r'))$;*
- *for every finite run r ending in (ℓ, ν), if $\nu + \mathsf{del}(\sigma(r)) \in (M_k, M_{k+1})$, we have $\nu + \mathsf{del}(\sigma(r)) \in (M_k, M_k + \eta] \cup [M_{k+1} - \eta, M_{k+1})$.*

We write UStrat_1^η and UStrat_2^η for the set of η-region-uniform strategies for Players 1 and 2. We also define upper-value $\overline{\mathsf{UVal}}^\eta(s)$ when both players are restricted to use only η-region-uniform strategies. Formally,

$$\overline{\mathsf{UVal}}^\eta(s) = \inf_{\sigma_1 \in \mathsf{UStrat}_1^\eta} \sup_{\sigma_2 \in \mathsf{UStrat}_2^\eta} \mathsf{Cost}(\mathsf{Play}(s, \sigma_1, \sigma_2)), \text{ for all } s \in S.$$

Example 1. Consider PTG $\mathcal{A}_1$ shown in Fig. 4 (that is not a 1BPTG since there are three distinct price-rates). A strategy of Player 2 is entirely described by the time spent in the initial location with initial valuation 0. For example, Player 2 can choose to delay 1/2 time units before jumping in the next location. Indeed, the lower and upper value of the game is $-\frac{1}{2}$. However, this strategy is not η-region-uniform. Instead, an η-region-uniform strategy will delay t time units with $t \in [0, \eta] \cup [1 - \eta, 1]$. Hence, the upper value when players can only use η-region-uniform strategies is equal to -1.

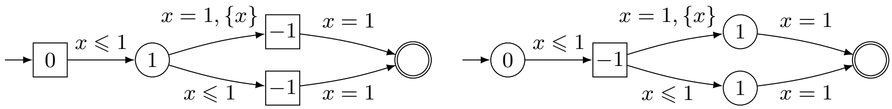

Fig. 4. The value in the left-side one-clock PTG $\mathcal{A}_1$ with price-rates in $\{-1, 0, 1\}$ is $-\frac{1}{2}$, while the value in the right-side PTG $\mathcal{A}_2$ is $\frac{1}{2}$

Contrary to this example, the next lemma shows that, in 1BPTGs, the upper value of the game increases when we restrict ourselves to η-region-uniform strategies. Intuitively, every cost that Player 2 can secure with general strategies, it can also secure it with η-region-uniform strategies against η-region-uniform strategies of Player 1.

Lemma 1. $\overline{\mathsf{Val}}(s) \leqslant \overline{\mathsf{UVal}}^{\eta}(s)$, *for every* 1BPTG $\mathcal{A}$, $s \in S$ *and* $\eta \in (0, \frac{1}{3})$,

5.2 Reduction to η-Convergent Strategies

A similar result concerning the lower values of the games can be shown in case of η-region-uniform strategies. In subsequent proofs, we need a stronger result to avoid situations detailed in Example 2, where player 2 needs infinite precision to play incrementally closer to borders (as well as an infinite memory). For this reason, we restrict the shape of strategies to force them to play at distance $\frac{\eta}{2^n}$ of borders when playing the nth round of the game. The slight asymmetry in the definitions for the two players is exploited in proving subsequent results.

Definition 6 (η-convergent strategies). *Let* $\eta \in (0, \frac{1}{3})$ *be a constant. A strategy* $\sigma \in \mathsf{Strat}_1 \cup \mathsf{Strat}_2$ *is said to be* η-convergent *if* σ *is* η*-region-uniform and for all finite run* r *of length* n *ending in* (ℓ, ν)*:*

- *if* $\sigma \in \mathsf{Strat}_1$*, there exists* k *such that either* $|\nu + \mathsf{del}(\sigma(r)) - M_k| \leqslant \frac{\eta}{2^{n+1}}$*, or* $\mathsf{del}(\sigma(r)) = 0$ *and* $\nu \in (M_k + \frac{\eta}{2^{n+1}}, M_k + \eta]$*;*
- *if* $\sigma \in \mathsf{Strat}_2$*, there exists* k *such that either* $\nu + \mathsf{del}(\sigma(r)) \in \{M_k + \frac{\eta}{2^{n+1}}\} \cup [M_k - \frac{\eta}{2^{n+1}}, M_k)$*, or* $\mathsf{del}(\sigma(r)) = 0$ *and* $\nu \in (M_k + \frac{\eta}{2^{n+1}}, M_k + \eta]$*.*

We let CStrat_1^{η} *and* CStrat_2^{η} *be respectively the set of* η*-convergent strategies for Player 1 and Player 2, and we define, for every configuration* $s \in S$*,* $\underline{\mathsf{CVal}}^{\eta}(s) = \sup_{\sigma_2 \in \mathsf{CStrat}_2^{\eta}} \inf_{\sigma_1 \in \mathsf{CStrat}_1^{\eta}} \mathsf{Cost}(\mathsf{Play}(s, \sigma_1, \sigma_2))$ *.*

Example 2. Consider the 1BPTG $\mathcal{A}_3$ composed of a vertex per player, on top of the target vertex. In its vertex, having price-rate 0, Player 1 must choose between going to the target vertex, or going to the vertex of Player 2 by resetting clock x. In its vertex, having price-rate -1, Player 2 must go back to the vertex of Player 1, with a guard $x > 0$: hence, Player 2 would like to exit as soon as possible, but because of the guard, he must spend some time before exiting. If Player 2 plays according to a finite-memory strategy, there must be a bound ε such that Player 2 always stays in his state for a duration bounded from below by ε, and Player 1 can exploit it by letting the game continue for an arbitrarily long time to achieve an arbitrarily small payoff. On the other hand, if Player 2 plays an infinite-memory η-convergent strategy by staying in his location for a duration $\varepsilon/2^n$ in his n-th visit to its location, Player 2 ensures a payoff $-\varepsilon$ for an arbitrarily small $\varepsilon > 0$, resulting in the value 0 of the game.

It is clear from the previous example that Player 2 needs infinite-memory strategies to optimize his objective. The following lemma formalizes our intuition that the lower value of the game decreases when we restrict ourselves to η-convergent strategies. Intuitively, every cost that Player 1 can secure with general strategies, it can also secure it with η-convergent strategies against an η-convergent strategy of Player 2.

Lemma 2. $\underline{\mathsf{CVal}}^\eta(s) \leqslant \underline{\mathsf{Val}}(s)$ *, for every* 1BPTG $\mathcal{A}$*,* $s \in S$ *and* $\eta \in (0, \frac{1}{3})$.

Observe that this lemma fails to hold when location price-rates can take more than two values as exemplified by arena $\mathcal{A}_2$ in Fig. 4. It shows a game with three distinct prices with lower and upper value equal to 1/2. However, when restricted to η-convergent strategies, the lower value equals 1.

Our next goal is to find a common bound being both a lower bound on $\underline{\mathsf{CVal}}^\eta(s)$ and an upper bound on $\overline{\mathsf{UVal}}^\eta(s)$ by studying the value of a reachability-cost game on a finitary abstraction of 1BPTGs.

5.3 Finite Abstraction of 1BPTGs

We now construct a finite price game graph $\tilde{\mathcal{A}}$ from any 1BPTG $\mathcal{A}$, as a finite abstraction of the infinite weighted game $[\![\mathcal{A}]\!]$, based on η-regions. Since we have learned that η-region-uniform strategies suffice, we limit ourselves to playing at a distance at most η from the borders of regions. Observe that only η-regions close to the borders are of interest, and moreover η-regions after the maximal constant M_K are not useful since $\mathcal{A}$ is bounded. Let $\mathcal{I}^\eta_\mathcal{A}$ be the set of remaining "useful" η-regions. For example, if constant appearing in the PTG are $M_1 = 2$ and $M_2 = 3$, we have $\mathcal{I}^\eta_\mathcal{A} = \{\{0\}, (0, \eta], [2-\eta, 2), \{2\}, (2, 2+\eta], [3-\eta, 3), \{3\}\}$. We next define the *delay* between two such η-regions $I \preceq J$, denoted by $d(I, J)$, as the closest integer of $q' - q$, where q (respectively, q') is the lower bound of interval I (respectively, J). For example, $d((2, 2+\eta], [3-\eta, 3)) = 1$ and $d(\{0\}, [2-\eta, 2)) = 2$.

Definition 7. *For every* 1BPTG $\mathcal{A}$ *we define its border abstraction as a finite priced game graph* $\tilde{\mathcal{A}} = (V = V_1 \uplus V_2, A, E, \pi, V_f)$ *where:*

- $V_i = \{(\ell, I) \mid \ell \in L_i, I \in \mathcal{I}^\eta_\mathcal{A}, I \subseteq \mathrm{Inv}(\ell)\}$ *for* $i \in \{1, 2\}$*;*
- $A = \mathcal{I}^\eta_\mathcal{A} \times \Sigma$*;*
- E *is the set of tuples* $((\ell, I), (J, a), (\ell', J'))$ *such that* $I \preceq J$ *and for all* $I \preceq K \preceq J$ *we have* $K \subseteq \mathrm{Inv}(\ell)$ *and* $J \subseteq \zeta$ *and* $J' = J[R := 0]$ *with* $(\zeta, R, \ell') = \delta(\ell, a)\}$*;*
- $\pi((\ell, I), (J, a), (\ell', J')) = \omega(\ell) \times d(I, J) + \omega(a)$*; and*
- $V_f = \{(\ell, I) \mid \ell \in L_f, I \in \mathcal{I}^\eta_\mathcal{A}\}$.

In a border abstraction game $\tilde{\mathcal{A}}$, the meaning of action (J, a) is that the player wants to let time elapse until it reaches the η-region J, then playing label a. It simulates any timed move (t, a) with t any delay reaching a point in J.

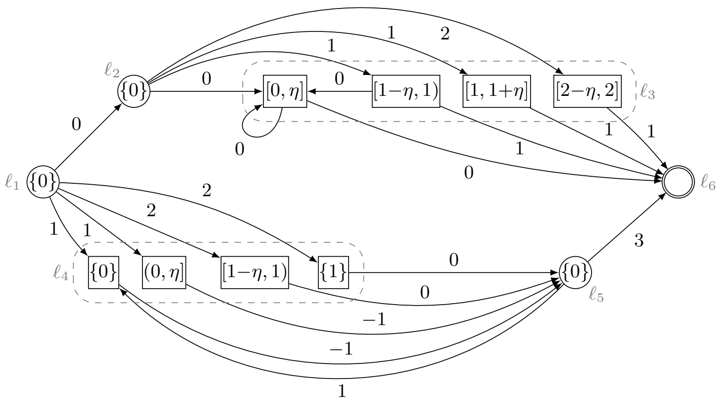

Fig. 5. Finite weighted game associated with the 1BPTG of Fig. 1

Example 3. Consider the border abstraction of the 1BPTG of Fig. 1 shown in Fig. 5. Observe that we depict only a succinct representation of the real abstraction, since we only show the reachable part of the game from $(\ell_1, 0)$, and we have removed multiple edges (introduced due to label hiding) and kept only the most useful ones for the corresponding player. For example, consider the location $(\ell_5, \{0\})$. There are edges labelled by (J, a) for every interval $J \in \mathcal{I}^{\eta}_{\mathcal{A}}$, all directed to $(\ell_4, \{0\})$ due to a reset being performed there. We only show the best possible edge—the one with lowest price—since location ℓ_5 belongs to Player 1, who seeks to minimise cost. Each vertex contains the η-region it represents. Thanks to Theorem 1, it is possible to compute the optimal value as well as optimal strategies for both players. Here, the value of state $(\ell_1, 0)$ is 1, and an optimal strategy for Player 1 is to follow action $(\{0\}, a)$ (i.e., jump to ℓ_2 immediately), and then action $(\{1\}, a)$ (i.e., to delay 1 time unit, before jumping in ℓ_3).

Lemma 3. *Let $\mathcal{A}$ be a* 1BPTG *and $\tilde{\mathcal{A}}$ be its border abstraction. Suppose that for all $0 \leqslant k \leqslant K$ and $\ell \in L$ we have that $\mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\}))$ is finite. Then, for all $\varepsilon > 0$, there is $\eta > 0$ s.t. $\overline{\mathsf{UVal}}^{\eta}_{\mathcal{A}}((\ell, M_k)) - \varepsilon \leqslant \mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\})) \leqslant \underline{\mathsf{CVal}}^{\eta}_{\mathcal{A}}((\ell, M_k)) + \varepsilon$.*

Combining this result with Theorem 1 we obtain the following.

Corollary 1. 1BPTG*s are determined and we can compute their values in pseudo-polynomial time. Moreover, in case the values are finite, ε-optimal strategies exist for both players: Player 2 may require infinite memory strategies, whereas finite memory is sufficient for Player 1. Finally, ε-optimal strategies can also be computed in pseudo-polynomial time.*

Proof. In case of infinite values $\mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\}))$, we can show directly that $\overline{\mathsf{Val}}_{\mathcal{A}}((\ell, M_k)) = \mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\})) = \underline{\mathsf{Val}}_{\mathcal{A}}((\ell, M_k))$. Otherwise, let $\varepsilon > 0$. By Lemma 3, we know that there exists $\eta > 0$ such that for every location $\ell \in L$ and integer $0 \leqslant k \leqslant K$:

$$\overline{\mathsf{UVal}}^{\eta}_{\mathcal{A}}((\ell, M_k)) - \varepsilon \leqslant \mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\})) \leqslant \underline{\mathsf{CVal}}^{\eta}_{\mathcal{A}}((\ell, M_k)) + \varepsilon\,.$$

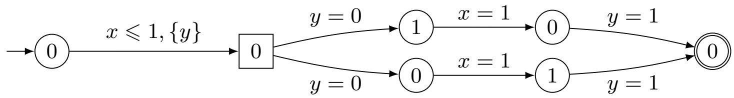

Fig. 6. A two-clock PTG with prices of locations in $\{0, +1\}$ and value 1/2

Moreover Lemma 1 and 2 show that:

$$\underline{\mathsf{CVal}}^\eta((\ell, M_k)) \leqslant \underline{\mathsf{Val}}((\ell, M_k)) \leqslant \overline{\mathsf{Val}}((\ell, M_k)) \leqslant \overline{\mathsf{UVal}}^\eta((\ell, M_k))\,.$$

Both inequalities combined permit to obtain

$$\mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\})) - \varepsilon \leqslant \underline{\mathsf{Val}}((\ell, M_k)) \leqslant \overline{\mathsf{Val}}((\ell, M_k)) \leqslant \mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\})) + \varepsilon\,.$$

Taking the limit when ε tends to 0, we obtain that $\underline{\mathsf{Val}}((\ell, M_k)) = \overline{\mathsf{Val}}((\ell, M_k)) = \mathsf{Val}_{\tilde{\mathcal{A}}}((\ell, \{M_k\}))$. Therefore, 1BPTG are determined. Moreover, in case of finite values, the proof of Lemma 3 permits to construct ε-optimal η-region-uniform strategies σ_1^* (with finite memory) and σ_2^* (which is moreover η-convergent). □

In the case of 1BPTGs, the finite values are integers. This property fails if we allow more than one clock, as shows Fig. 6 with a two-clock PTG with price-rates in $\{0, 1\}$ and optimal value $\frac{1}{2}$. It also fails if we allow more than two price-rates as was shown in Fig. 4. However for 1PTG(0, 1) with prices of labels in $\mathbb{N}$, the value of the game is necessarily nonnegative disallowing the case $-\infty$. The case $+\infty$ can be detected in polynomial time. If the value is not $+\infty$, the exact computation in the finite abstraction $\tilde{\mathcal{A}}$ can be performed in polynomial time (see [14] or [15]), resulting in a polynomial algorithm for PTGs. The sketch of Theorem 3 is now complete. Notice that our proof shows that optimal value functions (as defined in [10,20,15]) of such games have a polynomial number of line segments, and hence algorithms presented in [10,20,15] are indeed polynomial time.

6 Conclusion

We revisited games with reachability objective on PTGs with both positive and negative price-rates. We showed undecidability of all classes of constrained-price reachability objectives with two or more clocks. We also observed that adding bounded-time restriction does not recover decidability, even with nonnegative prices. We also partially answer the question regarding polynomial-time algorithm for one-clock PTGs by showing that for a bi-valued variant the problem is in pseudo-polynomial time. However, the existence of a polynomial-time algorithm for multi-priced one-clock PTGs with nonnegative price-rates, and the existence of algorithm for computing ε-optimal strategies for PTGs with arbitrary number of clocks remain open problems.

References

1. Alur, R., Bernadsky, M., Madhusudan, P.: Optimal reachability for weighted timed games. In: Díaz, J., Karhumäki, J., Lepistö, A., Sannella, D. (eds.) ICALP 2004. LNCS, vol. 3142, pp. 122–133. Springer, Heidelberg (2004)
2. Alur, R., Dill, D.L.: A theory of timed automata. TCS 126(2), 183–235 (1994)
3. Alur, R., La Torre, S., Pappas, G.J.: Optimal paths in weighted timed automata. TCS 318(3), 297–322 (2004)
4. Behrmann, G., Fehnker, A., Hune, T., Larsen, K.G., Pettersson, P., Romijn, J.M.T., Vaandrager, F.W.: Minimum-cost reachability for priced timed automata. In: Di Benedetto, M.D., Sangiovanni-Vincentelli, A.L. (eds.) HSCC 2001. LNCS, vol. 2034, pp. 147–161. Springer, Heidelberg (2001)
5. Berendsen, J., Chen, T., Jansen, D.N.: Undecidability of cost-bounded reachability in priced probabilistic timed automata. In: Chen, J., Cooper, S.B. (eds.) TAMC 2009. LNCS, vol. 5532, pp. 128–137. Springer, Heidelberg (2009)
6. Bouyer, P., Brihaye, T., Bruyère, V., Raskin, J.-F.: On the optimal reachability problem of weighted timed automata. FMSD 31(2), 135–175 (2007)
7. Bouyer, P., Brihaye, T., Markey, N.: Improved undecidability results on weighted timed automata. IPL 98(5), 188–194 (2006)
8. Bouyer, P., Brinksma, E., Larsen, K.G.: Optimal infinite scheduling for multi-priced timed automata. FMSD 32(1), 3–23 (2008)
9. Bouyer, P., Cassez, F., Fleury, E., Larsen, K.G.: Optimal strategies in priced timed game automata. In: Lodaya, K., Mahajan, M. (eds.) FSTTCS 2004. LNCS, vol. 3328, pp. 148–160. Springer, Heidelberg (2004)
10. Bouyer, P., Larsen, K.G., Markey, N., Rasmussen, J.I.: Almost optimal strategies in one-clock priced timed games. In: Arun-Kumar, S., Garg, N. (eds.) FSTTCS 2006. LNCS, vol. 4337, pp. 345–356. Springer, Heidelberg (2006)
11. Brihaye, T., Bruyère, V., Raskin, J.-F.: On optimal timed strategies. In: Pettersson, P., Yi, W. (eds.) FORMATS 2005. LNCS, vol. 3829, pp. 49–64. Springer, Heidelberg (2005)
12. Brihaye, T., Doyen, L., Geeraerts, G., Ouaknine, J., Raskin, J.-F., Worrell, J.: Time-bounded reachability for monotonic hybrid automata: Complexity and fixed points. In: Van Hung, D., Ogawa, M. (eds.) ATVA 2013. LNCS, vol. 8172, pp. 55–70. Springer, Heidelberg (2013)
13. Brihaye, T., Geeraerts, G., Krishna, S.N., Manasa, L., Monmege, B., Trivedi, A.: Reachability-cost games with negative weights. Technical report (2014), `http://arxiv.org/abs/1404.5894`
14. Brihaye, T., Geeraerts, G., Monmege, B.: Reachability-cost games with negative weights. Technical report (2014), `http://www.ulb.ac.be/di/verif/monmege/download/priced-games.pdf`
15. Hansen, T.D., Ibsen-Jensen, R., Miltersen, P.B.: A faster algorithm for solving one-clock priced timed games. In: D'Argenio, P.R., Melgratti, H. (eds.) CONCUR 2013 – Concurrency Theory. LNCS, vol. 8052, pp. 531–545. Springer, Heidelberg (2013)
16. Laroussinie, F., Markey, N., Schnoebelen, P.: Model checking timed automata with one or two clocks. In: Gardner, P., Yoshida, N. (eds.) CONCUR 2004. LNCS, vol. 3170, pp. 387–401. Springer, Heidelberg (2004)
17. Minsky, M.L.: Computation: Finite and infinite machines. Prentice-Hall, Inc. (1967)

18. Ouaknine, J., Worrell, J.: Towards a theory of time-bounded verification. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010. LNCS, vol. 6199, pp. 22–37. Springer, Heidelberg (2010)
19. Ramadge, P.J., Wonham, W.M.: The control of discrete event systems. IEEE 77, 81–98 (1989)
20. Rutkowski, M.: Two-player reachability-price games on single-clock timed automata. In: QAPL 2011. EPTCS, vol. 57, pp. 31–46 (2011)
21. Zwick, U., Paterson, M.: The complexity of mean payoff games on graphs. TCS 158, 343–359 (1996)