Reachability-Time Games on Timed Automata* (Extended Abstract)

Marcin Jurdziński** and Ashutosh Trivedi

Department of Computer Science, University of Warwick, UK

Abstract. In a reachability-time game, players Min and Max choose moves so that the time to reach a final state in a timed automaton is minimised or maximised, respectively. Asarin and Maler showed decidability of reachability-time games on strongly non-Zeno timed automata using a value iteration algorithm. This paper complements their work by providing a strategy improvement algorithm for the problem. It also generalizes their decidability result because the proposed strategy improvement algorithm solves reachability-time games on all timed automata. The exact computational complexity of solving reachability-time games is also established: the problem is EXPTIME-complete for timed automata with at least two clocks.

1 Introduction

Timed automata [3] are a fundamental formalism for modelling and analysis of real-time systems. They have a rich theory, mature modelling and verification tools (e.g., UPPAAL, Kronos), and have been successfully applied to numerous industrial case studies. Timed automata are finite automata augmented by a finite number of continuous real variables, which are called clocks because their values increase with time at unit rate. Every clock can be reset when a transition of the automaton is performed, and clock values can be compared to integers as a way to constrain availability of transitions. The fundamental reachability problem is PSPACE-complete for timed automata [3]. The natural optimization problems of minimizing and maximizing reachability time in timed automata are also in PSPACE [13].

The reachability (or optimal reachability-time) problems in timed automata are fundamental to the *verification* of (quantitative timing) properties of systems modelled by timed automata [3]. On the other hand, the problem of *control-program synthesis* for real-time systems can be cast as a two-player reachability (or optimal reachability-time) games, where the two players, say Min and Max, correspond to the "controller" and the "environment", respectively, and control-program synthesis corresponds to computing winning (or optimal) strategies for Min. In other words, for control-program synthesis, we need to generalize optimization problems to *competitive optimization* problems. Reachability games [5] and reachability-time games [4] on timed automata are decidable. The former problem is EXPTIME-complete, but the elegant result of Asarin and

* This research was supported in part by EPSRC project EP/E022030/1.

** Part of this work was done when the author visited the Isaac Newton Institute for Mathematical Sciences, Cambridge. Financial support from the Institute is gratefully acknowledged.

Maler [4] for reachability-time games is limited to the class of strongly non-Zeno timed automata and no upper complexity bounds are given. A recent result of Henzinger and Prabhu [15] is that values of reachability-time games can be approximated for all timed automata, but computatability of the exact values was left open.

Our contribution. We show that exact values of reachability-time games on arbitrary timed automata are uniformly computable; here uniformity means that the output of our algorithm allows us, for every starting state, to compute efficiently the value of the game starting from this state. Unlike the paper of Asarin and Maler [4], we do not require timed automata to be strongly non-Zeno. We also establish the exact complexity of reachability and reachability-time games: they are EXPTIME-complete and two clocks are sufficient for EXPTIME-hardness. For the latter result, we reduce from a recently discovered EXPTIME-complete problem of countdown games [16].

We believe that an important contribution of this paper are the novel proof techniques used. We characterize the values of the game by *optimality equations* and then we use *strategy improvement* to solve them. This allows us to obtain an elementary and constructive proof of the fundamental determinacy result for reachability-time games, which at the same time yields an efficient algorithm matching the EXPTIME lower bound for the problem. Those techniques were known for finite state systems [17, 19] but we are not aware of any earlier algorithmic results based on optimality equations and strategy improvement for real-time systems such as timed automata.

Related and future work. A recent, concurrent, and independent work [12] establishes decidability of slightly different and more challenging reachability-time games "with the element of surprise" [14, 15]. In our model of timed games, players take turns to take unilateral decisions about the duration and type of subsequent game moves. Games with surprise are more challenging in two ways: in every round of the game, players have a "time race" to be the first to perform a move; moreover, players are forbidden to use strategies which "stop the time," because such strategies are arguably physically unrealistic and result in Zeno runs. In our reachability-time games, player Max may use such Zeno strategies in order to prevent reaching a final state. We conjecture that our techniques can be generalized to prevent player Max from using Zeno strategies.

A generalization of timed automata to priced (or weighted) timed automata [7] allows a rich variety of applications, e.g., to scheduling [6, 1, 18]. While the fundamental minimum reachability-price problem is PSPACE-complete [6, 8], the two-player reachability-price games are undecidable on priced timed automata with at least three clocks [9]. The reachability-price games are, however, decidable for priced timed automata with one clock [11], and on the class of strongly price-non-Zeno timed automata [2, 10]. Future work should include adapting the techniques of optimality equations and strategy improvement to (competitive) optimization problems on priced timed automata.

2 Timed Automata and Reachability-Time Games

We assume that, wherever appropriate, sets $\mathbb{N}$ of non-negative integers and $\mathbb{R}$ of reals contain a maximum element ∞, and we write $\mathbb{N}_{>0}$ for the set of positive integers and $\mathbb{R}_{\geq 0}$ for the set of non-negative reals. For $n \in \mathbb{N}$, we write $[\![n]\!]_{\mathbb{N}}$ for the set $\{0, 1, \ldots, n\}$, and $[\![n]\!]_{\mathbb{R}}$ for the set $\{r \in \mathbb{R} \,:\, 0 \le r \le n\}$ of non-negative reals bounded by n. For sets X and Y, we write $[X \to Y]$ for the set of functions $F : X \to Y$, and $[X \rightharpoonup Y]$ for the set of partial functions $F : X \rightharpoonup Y$.

Timed automata. Fix a constant $k \in \mathbb{N}$ for the rest of this paper. Let C be a finite set of *clocks*. A (k-bounded) *clock valuation* is a function $\nu : C \to [\![k]\!]_{\mathbb{R}}$; we write V for the set $[C \to [\![k]\!]_{\mathbb{R}}]$ of clock valuations. If $\nu \in V$ and $t \in \mathbb{R}_{\ge 0}$ then we write $\nu + t$ for the clock valuation defined by $(\nu + t)(c) = \nu(c) + t$, for all $c \in C$. For a set $C' \subseteq C$ of clocks and a clock valuation $\nu : C \to \mathbb{R}_{\ge 0}$, we define $\text{Reset}(\nu, C')(c) = 0$ if $c \in C'$, and $\text{Reset}(\nu, C')(c) = \nu(c)$ if $c \notin C'$.

The set of *clock constraints* over the set of clocks C is the set of conjunctions of *simple clock constraints*, which are constraints of the form $c \bowtie i$ or $c - c' \bowtie i$, where $c, c' \in C$, $i \in [\![k]\!]_{\mathbb{N}}$, and $\bowtie \in \{ <, >, =, \le, \ge \}$. There are finitely many simple clock constraints. For every clock valuation $\nu \in V$, let $\text{SCC}(\nu)$ be the set of simple clock constraints which hold in $\nu \in V$. A *clock region* is a maximal set $P \subseteq V$, such that for all $\nu, \nu' \in P$, we have $\text{SCC}(\nu) = \text{SCC}(\nu')$. In other words, every clock region is an equivalence class of the indistinguishability-by-clock-constraints relation, and vice versa. Note that ν and ν' are in the same clock region iff all clocks have the same integer parts in ν and ν', and if the partial orders of the clocks, determined by their fractional parts in ν and ν', are the same. For all $\nu \in V$, we write $[\nu]$ for the clock region of ν.

A *clock zone* is a convex set of clock valuations, which is a union of a set of clock regions. Note that a set of clock valuations is a zone iff it is definable by a clock constraint. For $W \subseteq V$, we write $\overline{W}$ for the smallest closed set in V which contains W. Observe that for every clock zone W, the set $\overline{W}$ is also a clock zone.

Let L be a finite set of *locations*. A *configuration* is a pair (ℓ, ν), where $\ell \in L$ is a location and $\nu \in V$ is a clock valuation; we write Q for the set of configurations. If $s = (\ell, \nu) \in Q$ and $c \in C$, then we write $s(c)$ for $\nu(c)$. A *region* is a pair (ℓ, P), where ℓ is a location and P is a clock region. If $s = (\ell, \nu)$ is a configuration then we write $[s]$ for the region $(\ell, [\nu])$. We write $\mathcal{R}$ for the set of regions. A set $Z \subseteq Q$ is a *zone* if for every $\ell \in L$, there is a clock zone W_ℓ (possibly empty), such that $Z = \{(\ell, \nu) \,:\, \ell \in L \text{ and } \nu \in W_\ell\}$. For a region $R = (\ell, P) \in \mathcal{R}$, we write $\overline{R}$ for the zone $\{(\ell, \nu) \,:\, \nu \in \overline{P}\}$.

A *timed automaton* $\mathcal{T} = (L, C, S, A, E, \delta, \rho, F)$ consists of a finite set of locations L, a finite set of clocks C, a set of *states* $S \subseteq Q$, a finite set of *actions* A, an *action enabledness function* $E : A \to 2^S$, a *transition function* $\delta : L \times A \to L$, a *clock reset function* $\rho : A \to 2^C$, and a set of *final states* $F \subseteq S$. We require that S, F, and $E(a)$ for all $a \in A$, are zones.

Clock zones, from which zones S, F, and $E(a)$, for all $a \in A$, are built, are typically specified by clock constraints. Therefore, when we consider a timed automaton as an input of an algorithm, its size should be understood as the sum of sizes of encodings of L, C, A, δ, and ρ, and the sizes of encodings of clock constraints defining zones S, F, and $E(a)$, for all $a \in A$. Our definition of a timed automaton may appear to differ from the usual ones [3, 7]. The differences are, however, superficial and mostly syntactic.

For a configuration $s = (\ell, \nu) \in Q$ and $t \in \mathbb{R}_{\ge 0}$, we define $s + t$ to be the configuration $s' = (\ell, \nu + t)$ if $\nu + t \in V$, and we then write $s \rightharpoonup_t s'$. We write $s \to_t s'$ if $s \rightharpoonup_t s'$ and for all $t' \in [0, t]$, we have $(\ell, s + t') \in S$. For an action $a \in A$, we define $\text{Succ}(s, a)$

to be the configuration $s' = (\ell', \nu')$, where $\ell' = \delta(\ell, a)$ and $\nu' = \text{Reset}(\nu, \rho(a))$, and we then write $s \stackrel{a}{\rightharpoonup} s'$. We write $s \xrightarrow{a} s'$ if $s \stackrel{a}{\rightharpoonup} s'$; $s, s' \in S$; and $s \in E(a)$. For technical convenience, and without loss of generality, we will assume throughout that for every $s \in S$, there exists $a \in A$, such that $s \xrightarrow{a} s'$.

For $s, s' \in S$, we say that s' is in the future of s, or equivalently, that s is in the past of s', if there is $t \in \mathbb{R}_{\geq 0}$, such that $s \to_t s'$; we then write $s \to_* s'$. For $R, R' \in \mathcal{R}$, we say that R' is in the future of R, or that R is in the past of R', if for all $s \in R$, there is $s' \in R'$, such that s' is in the future of s; we then write $R \to_* R'$. We say that R' is the *time successor* of R if $R \to_* R'$, $R \neq R'$, and for every $R'' \in \mathcal{R}$, we have that $R \to_* R'' \to_* R'$ implies $R'' = R$ or $R'' = R'$; we then write $R \to_{+1} R'$ or $R' \leftarrow_{+1} R$. Similarly, for $R, R' \in \mathcal{R}$, we write $R \xrightarrow{a} R'$ if there is $s \in R$, and there is $s' \in R'$, such that $s \xrightarrow{a} s'$.

We say that a region $R \in \mathcal{R}$ is *thin* if for every $s \in R$ and every $\varepsilon > 0$, we have that $[s] \neq [s + \varepsilon]$; other regions are called *thick*. We write $\mathcal{R}_{\text{Thin}}$ and $\mathcal{R}_{\text{Thick}}$ for the sets of thin and thick regions, respectively. Note that if $R \in \mathcal{R}_{\text{Thick}}$ then for every $s \in R$, there is an $\varepsilon > 0$, such that $[s] = [s + \varepsilon]$. Observe also, that the time successor of a thin region is thick, and vice versa.

A *timed action* is a pair $\tau = (a, t) \in A \times \mathbb{R}_{\geq 0}$. For $s \in Q$, we define $\text{Succ}(s, \tau) = \text{Succ}(s, (a, t))$ to be the configuration $s' = \text{Succ}(s + t, a)$, i.e., such that $s \rightharpoonup_t s'' \stackrel{a}{\rightharpoonup} s'$, and we then write $s \stackrel{a}{\rightharpoonup}_t s'$. We write $s \xrightarrow{a}_t s'$ if $s \to_t s'' \xrightarrow{a} s'$. If $\tau = (a, t)$ then we write $s \stackrel{\tau}{\rightharpoonup} s'$ instead of $s \stackrel{a}{\rightharpoonup}_t s'$, and $s \xrightarrow{\tau} s'$ instead of $s \xrightarrow{a}_t s'$.

A finite run of a timed automaton is a sequence $\langle s_0, \tau_1, s_1, \tau_2, \ldots, \tau_n, s_n \rangle \in S \times ((A \times \mathbb{R}_{\geq 0}) \times S)^*$, such that for all i, $1 \leq i \leq n$, we have $s_{i-1} \xrightarrow{\tau_i} s_i$. For a finite run $r = \langle s_0, \tau_1, s_1, \tau_2, \ldots, \tau_n, s_n \rangle$, we define $\text{Length}(r) = n$, and we define $\text{Last}(r) = s_n$ to be the state in which the run ends. We write Runs_{fin} for the set of finite runs. An infinite run of a timed automaton is a sequence $r = \langle s_0, \tau_1, s_1, \tau_2, \ldots \rangle$, such that for all $i \geq 1$, we have $s_{i-1} \xrightarrow{\tau_i} s_i$. For an infinite run r, we define $\text{Length}(r) = \infty$. For a run $r = \langle s_0, \tau_1, s_1, \tau_2, \ldots \rangle$, we define $\text{Stop}(r) = \inf\{i : s_i \in F\}$ and $\text{Time}(r) = \sum_{i=1}^{\text{Length}(r)} t_i$. We define $\text{ReachTime}(r) = \sum_{i=1}^{\text{Stop}(r)} t_i$ if $\text{Stop}(r) < \infty$, and $\text{ReachTime}(r) = \infty$ if $\text{Stop}(r) = \infty$, where for all $i \geq 1$, we have $\tau_i = (a_i, t_i)$.

Strategies. A reachability-time game Γ is a triple $(\mathcal{T}, L_{\text{Min}}, L_{\text{Max}})$, where $\mathcal{T}$ is a timed automaton $(L, C, S, A, E, \delta, \rho, F)$ and $(L_{\text{Min}}, L_{\text{Max}})$ is a partition of L. We define sets $Q_{\text{Min}} = \{(\ell, \nu) \in Q : \ell \in L_{\text{Min}}\}$, $Q_{\text{Max}} = Q \setminus Q_{\text{Min}}$, $S_{\text{Min}} = S \cap Q_{\text{Min}}$, $S_{\text{Max}} = S \setminus S_{\text{Min}}$, $\mathcal{R}_{\text{Min}} = \{[s] \,:\, s \in Q_{\text{Min}}\}$, and $\mathcal{R}_{\text{Max}} = \mathcal{R} \setminus \mathcal{R}_{\text{Min}}$.

A *strategy* for Min is a function $\mu : \text{Runs}_{\text{fin}} \to A \times \mathbb{R}_{\geq 0}$, such that if $\text{Last}(r) = s \in S_{\text{Min}}$ and $\mu(r) = \tau$ then $s \xrightarrow{\tau} s'$. Similarly, a strategy for Max is a function $\chi : \text{Runs}_{\text{fin}} \to A \times \mathbb{R}_{\geq 0}$, such that if $\text{Last}(r) = s \in S_{\text{Max}}$ and $\chi(r) = \tau$ then $s \xrightarrow{\tau} s'$. We write Σ_{Min} and Σ_{Max} for the sets of strategies for Min and Max, respectively. If players Min and Max use strategies μ and χ, respectively, then the (μ, χ)-run from a state s is the unique run $\text{Run}(s, \mu, \chi) = \langle s_0, \tau_1, s_1, \tau_2, \ldots \rangle$, such that $s_0 = s$, and for every $i \geq 1$, if $s_i \in S_{\text{Min}}$ or $s_i \in S_{\text{Max}}$, then $\mu(\text{Run}_i(s, \mu, \chi)) = \tau_{i+1}$ or $\chi(\text{Run}_i(s, \mu, \chi)) = \tau_{i+1}$, respectively, where $\text{Run}_i(s, \mu, \chi) = \langle s_0, \tau_1, s_1, \ldots, s_{i-1}, \tau_i, s_i \rangle$.

We say that a strategy μ for Min is *positional* if for all finite runs $r, r' \in \text{Runs}_{\text{fin}}$, we have that $\text{Last}(r) = \text{Last}(r')$ implies $\mu(r) = \mu(r')$. A positional strategy for Min can be then represented as a function $\mu : S_{\text{Min}} \to A \times \mathbb{R}_{\geq 0}$, which uniquely determines the strategy $\mu^\infty \in \Sigma_{\text{Min}}$ as follows: $\mu^\infty(r) = \mu(\text{Last}(r))$, for all finite runs $r \in \text{Runs}_{\text{fin}}$. Positional strategies for Max are defined and represented in the analogous way. We write Π_{Min} and Π_{Max} for the sets of positional strategies for Min and for Max, respectively.

Value of reachability-time game. For every $s \in S$, we define its *upper* and *lower values* by $\text{Val}^*(s) = \inf_{\mu \in \Sigma_{\text{Min}}} \sup_{\chi \in \Sigma_{\text{Max}}} \text{ReachTime}(\text{Run}(s, \mu, \chi))$, and $\text{Val}_*(s) = \sup_{\chi \in \Sigma_{\text{Max}}} \inf_{\mu \in \Sigma_{\text{Min}}} \text{ReachTime}(\text{Run}(s, \mu, \chi))$, respectively. The inequality $\text{Val}_*(s) \leq \text{Val}^*(s)$ always holds.

A reachability-time game is *determined* if for every $s \in S$, the lower and upper values are equal to each other; then the *value* $\text{Val}(s) = \text{Val}_*(s) = \text{Val}^*(s)$ is defined. For a strategy $\mu \in \Sigma_{\text{Min}}$, we define its value $\text{Val}^\mu(s) = \sup_{\chi \in \Sigma_{\text{Min}}} \text{ReachTime}(\text{Run}(s, \mu, \chi))$, and for $\chi \in \Sigma_{\text{Max}}$, we define $\text{Val}_\chi(s) = \inf_{\mu \in \Sigma_{\text{Min}}} \text{ReachTime}(\text{Run}(s, \mu, \chi))$. For an $\varepsilon > 0$, we say that a strategy $\mu \in \Sigma_{\text{Min}}$ or $\chi \in \Sigma_{\text{Max}}$ is ε-*optimal* if for every $s \in S$, we have $\text{Val}^\mu(s) \leq \text{Val}(s) + \varepsilon$ or $\text{Val}_\chi(s) \geq \text{Val}(s) - \varepsilon$, respectively. Note that if a game is determined then for every $\varepsilon > 0$, both players have ε-optimal strategies.

We say that a reachability-time game is *positionally determined* if for every $s \in S$, and for every $\varepsilon > 0$, both players have *positional* ε-optimal strategies from s. Our results (Lemma 1, Theorem 2, and Theorem 4) yield a constructive proof of the following fundamental result for reachability-time games.

Theorem 1 (Positional determinacy). *Reachability-time games are positionally determined.*

Optimality equations $\text{Opt}_{\text{MinMax}}(\Gamma)$. Our principal technique is to characterize the values $\text{Val}(s)$, for all $s \in S$, as solutions of an infinite system of *optimality equations*, and then to study the equations. We write $(T, D) \models \text{Opt}_{\text{MinMax}}(\Gamma)$, and we say that (T, D) is a solution of *optimality equations* $\text{Opt}_{\text{MinMax}}(\Gamma)$, if for all $s \in S$, we have:

- if $D(s) = \infty$ then $T(s) = \infty$; and
- if $s \in F$ then $(T(s), D(s)) = (0, 0)$;
- if $s \in S_{\text{Min}} \setminus F$, then $T(s) = \inf_{a,t}\{t + T(s') : s \xrightarrow{a}_t s'\}$, and $D(s) = \min\big\{1 + d' : T(s) = \inf_{a,t}\{t + T(s') : s \xrightarrow{a}_t s' \text{ and } D(s') = d'\}\big\}$; and
- if $s \in S_{\text{Max}} \setminus F$, then $T(s) = \sup_{a,t}\{t + T(s') : s \xrightarrow{a}_t s'\}$, and $D(s) = \max\big\{1 + d' : T(s) = \sup_{a,t}\{t + T(s') : s \xrightarrow{a}_t s' \text{ and } D(s') = d'\}\big\}$.

Intuitively, in the equations above, $T(s)$ and $D(s)$ capture "optimal time to reach a final state" and "optimal distance to reach a final state in optimal time" from state $s \in S$, respectively. The following key lemma establishes that in order to solve a reachability-time game Γ, it suffices to find a solution of $\text{Opt}_{\text{MinMax}}(\Gamma)$.

Lemma 1 (ε-Optimal strategies from optimality equations). *If $T : S \to \mathbb{R}$ and $D : S \to \mathbb{N}$ are such that $(T, D) \models \text{Opt}_{\text{MinMax}}(\Gamma)$, then for all $s \in S$, we have $\text{Val}(s) = T(s)$ and for every $\varepsilon > 0$, both players have* positional *ε-optimal strategies.*

3 Timed Region Graph

In this section we argue that the task of solving $\mathrm{Opt}_{\mathrm{MinMax}}(\Gamma)$ can be reduced to solving a simpler system of equations $\mathrm{Opt}_{\mathrm{MinMax}}(\widehat{\Gamma})$, which is also infinite, but whose right-hand sides are minima or maxima of only finitely many expressions.

Simple timed actions. Define the finite set of *simple timed actions* $\mathcal{A} = A \times [\![k]\!]_{\mathbb{N}} \times C$. For $s \in Q$ and $\alpha = (a, b, c) \in \mathcal{A}$, we define $t(s, \alpha) = b - s(c)$ if $s(c) \leq b$, and $t(s, \alpha) = 0$ if $s(c) > b$; and we define $\mathrm{Succ}(s, \alpha)$ to be the state $s' = \mathrm{Succ}(s, \tau(\alpha))$, where $\tau(\alpha) = (a, t(s, \alpha))$; we then write $s \xrightarrow{\alpha} s'$. We also write $s \xrightarrow{\alpha} s'$ if $s \xrightarrow{\tau(\alpha)} s'$. Note that if $\alpha \in \mathcal{A}$ and $s \xrightarrow{\alpha} s'$ then $[s'] \in \mathcal{R}_{\mathrm{Thin}}$. Observe that for every thin region $R' \in \mathcal{R}_{\mathrm{Thin}}$, there is a number $b \in [\![k]\!]_{\mathbb{N}}$ and a clock $c \in C$, such that for every $R \in \mathcal{R}$ in the past of R', we have that $s \in R$ implies $(s + (b - s(c)) \in R'$; we then write $R \rightarrow_{b,c} R'$. For $\alpha = (a, b, c) \in \mathcal{A}$ and $R, R' \in \mathcal{R}$, we write $R \xrightarrow{\alpha} R'$ or $R \xrightarrow{a}_{b,c} R'$, if $R \rightarrow_{b,c} R'' \xrightarrow{a} R'$, for some $R'' \in \mathcal{R}_{\mathrm{Thin}}$.

Timed region graph $\widehat{\Gamma}$. Let $\Gamma = (\mathcal{T}, L_{\mathrm{Min}}, L_{\mathrm{Max}})$ be a reachability-time game. We define the *timed region graph* $\widehat{\Gamma}$ to be the finite edge-labelled graph $(\mathcal{R}, \mathcal{M})$, where the set $\mathcal{R}$ of regions of timed automaton $\mathcal{T}$ is the set of vertices, and the labelled edge relation $\mathcal{M} \subseteq \mathcal{R} \times \mathcal{A} \times \mathcal{R}$ is defined in the following way. For $\alpha = (a, b, c) \in \mathcal{A}$ and $R, R' \in \mathcal{R}$ we have $(R, \alpha, R') \in \mathcal{M}$, sometimes denoted by $R \overset{\alpha}{\rightsquigarrow} R'$, if and only if one of the following conditions holds:

– there is an $R'' \in \mathcal{R}$, such that $R \rightarrow_{b,c} R'' \xrightarrow{a} R'$; or
– $R \in \mathcal{R}_{\mathrm{Min}}$, and there are $R'', R''' \in \mathcal{R}$, such that $R \rightarrow_{b,c} R'' \rightarrow_{+1} R''' \xrightarrow{a} R'$; or
– $R \in \mathcal{R}_{\mathrm{Max}}$, and there are $R'', R''' \in \mathcal{R}$, such that $R \rightarrow_{b,c} R'' \leftarrow_{+1} R''' \xrightarrow{a} R'$.

Observe that in all the cases above we have that $R'' \in \mathcal{R}_{\mathrm{Thin}}$ and $R''' \in \mathcal{R}_{\mathrm{Thick}}$. The motivation for the second case is the following. Let $R \rightarrow_{*} R''' \xrightarrow{a} R'$, where $R \in \mathcal{R}_{\mathrm{Min}}$ and $R''' \in \mathcal{R}_{\mathrm{Thick}}$. One of the key results that we establish is that in a state $s \in R$, among all $t \in \mathbb{R}_{\geq 0}$, such that $s + t \in R'''$, the smaller the t, the "better" the timed action (a, t) is for player Min. Note, however, that the set $\{t \in \mathbb{R}_{\geq 0} \ : \ s + t \in R'''\}$ is an open interval because $R''' \in \mathcal{R}_{\mathrm{Thick}}$, and hence it does not have the smallest element. Therefore, for every $s \in R$, we model the "best" time to wait, when starting from s, before performing an a-labelled transition from region R''' to region R', by taking the infimum of the set $\{t \in \mathbb{R}_{\geq 0} \ : \ s + t \in R'''\}$. Observe that this infimum is equal to the $t_{R''} \in \mathbb{R}_{\geq 0}$, such that $s + t_{R''} \in R''$, where $R'' \rightarrow_{+1} R'''$, and that $t_{R''} = b - s(c)$, where $R \rightarrow_{b,c} R''$. In the timed region graph $\widehat{\Gamma}$, we summarize this model of the "best" timed action from region R to region R' via region R''', by having a move $(R, \alpha, R') \in \mathcal{M}$, where $\alpha = (a, b, c)$. The motivation for the first and the third cases of the definition of $\mathcal{M}$ is similar.

Regional functions and optimality equations $\mathrm{Opt}_{\mathrm{MinMax}}(\widehat{\Gamma})$. Recall from Section 2 that a solution of optimality equations $\mathrm{Opt}_{\mathrm{MinMax}}(\Gamma)$ for a reachability-time game Γ is a pair of functions (T, D), such that $T : S \to \mathbb{R}$ and $D : S \to \mathbb{N}$. Our goal is to define analogous optimality equations $\mathrm{Opt}_{\mathrm{MinMax}}(\widehat{\Gamma})$ for the timed region graph $\widehat{\Gamma}$.

If $R \overset{\alpha}{\rightsquigarrow} R'$, where $R, R' \in \mathcal{R}$ and $\alpha \in \mathcal{A}$, then $s \in R$ does not imply that $\mathrm{Succ}(s, \alpha) \in R'$; however, $s \in R$ implies $\mathrm{Succ}(s, \alpha) \in \overline{R'}$. In order to correctly capture the constraints for successor states which fall out of the “target” region R' of a move of the form $R \overset{\alpha}{\rightsquigarrow} R'$, we consider, as solutions of optimality equations $\text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$, *regional functions* of types $T : \mathcal{R} \to [S \rightharpoonup \mathbb{R}]$ and $D : \mathcal{R} \to [S \rightharpoonup \mathbb{N}]$, where for every $R \in \mathcal{R}$, the domain of partial functions $T(R)$ and $D(R)$ is $\overline{R}$. Sometimes, when defining a regional function $F : \mathcal{R} \to [S \rightharpoonup \mathbb{R}]$, it will only be natural to define $F(R)$ for all $s \in R$, instead of all $s \in \overline{R}$. This is not a problem, however, because defining $F(R)$ on the region R uniquely determines the continuous extension of $F(R)$ to $\overline{R}$. For a function $F : \mathcal{R} \to [S \rightharpoonup \mathbb{R}]$, we define the function $\widetilde{F} : S \to \mathbb{R}$ by $\widetilde{F}(s) = F([s])(s)$.

If $F, F', G, G' : S \to \mathbb{R}$ then we write $F \le F'$ or $(F, G) \le^{\text{lex}} (F', G')$, if for all $s \in S$, we have $F(s) \le F'(s)$ or $(F(s), G(s)) \le^{\text{lex}} (F'(s), G'(s))$, respectively, where $\le^{\text{lex}}$ is the lexicographic order. Moreover, $F < F'$ or $(F, G) <^{\text{lex}} (F', G')$, if $F \le F'$ or $(F, G) \le^{\text{lex}} (F', G')$, and there is $s \in S$, such that $F(s) < F'(s)$ or $(F(s), G(s)) <^{\text{lex}} (F'(s), G'(s))$, respectively.

If $\alpha \in \mathcal{A}$; $R, R' \in \mathcal{R}$; $R \xrightarrow{\alpha} R'$; and $T : R' \to \mathbb{R}$ and $D : R' \to \mathbb{N}$, then we define the functions $T^{\oplus}_{\alpha} : R \to \mathbb{R}$ and $D^{\boxplus}_{\alpha} : R \to \mathbb{R}$, by $T^{\oplus}_{\alpha}(s) = t(s, \alpha) + T(\text{Succ}(s, \alpha))$ and $D^{\boxplus}_{\alpha}(s) = 1 + D(\text{Succ}(s, \alpha))$, for all $s \in R$. We write $(T, D) \models \text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$ if for all $s \in S$, we have the following:

- if $\widetilde{D}(s) = \infty$ then $\widetilde{T}(s) = \infty$; and $\big(\widetilde{T}(s), \widetilde{D}(s)\big) = (0, 0)$ if $s \in F$;
- $\big(\widetilde{T}(s), \widetilde{D}(s)\big) = \min^{\text{lex}}{}_{m \in \mathcal{M}} \big\{\big(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\big) \ : \ m = ([s], \alpha, R')\big\}$ if $s \in S_{\text{Min}} \setminus F$; and
- $\big(\widetilde{T}(s), \widetilde{D}(s)\big) = \max^{\text{lex}}{}_{m \in \mathcal{M}} \big\{\big(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\big) \ : \ m = ([s], \alpha, R')\big\}$ if $s \in S_{\text{Max}} \setminus F$.

Solutions of $\text{Opt}_{\text{MinMax}}(\Gamma)$ from solutions of $\text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$. In this subsection we show that the function $(T, D) \mapsto (\widetilde{T}, \widetilde{D})$ translates solutions of $\text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$ to solutions of $\text{Opt}_{\text{MinMax}}(\Gamma)$. In other words, the function $\Gamma \mapsto \widehat{\Gamma}$ is a reduction from the problem of computing values in reachability-time games to the problem of solving optimality equations for timed region graphs.

In order to prove correctness of the reduction, however, we need extra properties of the solution (T, D) of $\text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$, namely that T is regionally simple, and that D is regionally constant. Let $X \subseteq Q$. A function $T : X \to \mathbb{R}$ is *simple* [4] if either: there is $e \in \mathbb{Z}$, such that for every $s \in X$, we have $T(s) = e$; or there are $e \in \mathbb{Z}$ and $c \in C$, such that for every $s \in X$, we have $T(s) = e - s(c)$. Observe that if $R \in \mathcal{R}$ and $T : R \to \mathbb{R}$ is simple, then the unique continuous extension of T to $\overline{R}$ is also simple. We say that a function $F : \mathcal{R} \to [S \rightharpoonup \mathbb{R}]$ is *regionally simple* or *regionally constant*, if for every region $R \in \mathcal{R}$, the function $F(R) : \overline{R} \to \mathbb{R}$ is simple or constant, respectively.

Theorem 2 (Correctness of reduction to timed region graphs). *If $T : S \to \mathbb{R}$ and $D : S \to \mathbb{N}$ are such that $(T, D) \models \text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$, T is regionally simple, and D is regionally constant, then $(\widetilde{T}, \widetilde{D}) \models \text{Opt}_{\text{MinMax}}(\Gamma)$.*

4 Solving Optimality Equations by Strategy Improvement

In this section we give a strategy improvement algorithm to compute a solution (T, D) of $\text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$, and we argue that it satisfies the assumptions of Theorem 2, i.e., T is regionally simple and D is regionally constant.

Positional strategies. A positional strategy for player Max in a timed region graph $\widehat{\Gamma}$ is a function $\chi : S_{\text{Max}} \to \mathcal{M}$, such that for every $s \in S_{\text{Max}}$, we have $\chi(s) = ([s], \alpha, R)$, for some $\alpha \in \mathcal{A}$ and $R \in \mathcal{R}$. A strategy $\chi : S_{\text{Max}} \to \mathcal{M}$ is *regionally constant* if for all $s, s' \in S_{\text{Max}}$, we have that $[s] = [s']$ implies $\chi(s) = \chi(s')$; we can then write $\chi([s])$ for $\chi(s)$. Positional strategies for player Min are defined analogously. We write Δ_{Max} and Δ_{Min} for the sets of positional strategies for players Max and Min, respectively.

If $\chi \in \Delta_{\text{Max}}$ is regionally constant then we define the strategy subgraph $\widehat{\Gamma}\restriction\chi$ to be the subgraph $(\mathcal{R}, \mathcal{M}_\chi)$ where $\mathcal{M}_\chi \subseteq \mathcal{M}$ consists of: all moves $(R, \alpha, R') \in \mathcal{M}$, such that $R \in \mathcal{R}_{\text{Min}}$; and of all moves $m = (R, \alpha, R')$, such that $R \in \mathcal{R}_{\text{Max}}$ and $\chi(R) = m$. The strategy subgraph $\widehat{\Gamma}\restriction\mu$ for a regionally constant positional strategy $\mu \in \Delta_{\text{Min}}$ for player Min is defined analogously. We say that $R \in \mathcal{R}$ is *choiceless* in a timed region graph $\widehat{\Gamma}$ if R has a unique successor in $\widehat{\Gamma}$. We say that $\widehat{\Gamma}$ is 0-player if all $R \in \mathcal{R}$ are choiceless in $\widehat{\Gamma}$; we say that $\widehat{\Gamma}$ is 1-player if either all $R \in \mathcal{R}_{\text{Min}}$ or all $R \in \mathcal{R}_{\text{Max}}$ are choiceless in $\widehat{\Gamma}$; every timed region graph $\widehat{\Gamma}$ is 2-player. Note that if χ and μ are positional strategies in $\widehat{\Gamma}$ for players Max and Min, respectively, then $\widehat{\Gamma}\restriction\chi$ and $\widehat{\Gamma}\restriction\mu$ are 1-player and $(\widehat{\Gamma}\restriction\chi)\restriction\mu$ is 0-player.

For functions $T : \mathcal{R} \to [S \to \mathbb{R}]$ and $D : \mathcal{R} \to [S \to \mathbb{R}]$, and $s \in S_{\text{Max}}$, we define sets $M^*(s, (T, D))$ and $M_*(s, (T, D))$, respectively, of moves enabled in s which are (lexicographically) (T, D)-optimal for player Max and Min, respectively:

$$M^*(s,(T,D)) = \operatorname*{argmax}_{m\in\mathcal{M}}{}^{\text{lex}} \left\{ \left(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\right) \,:\, m = ([s],\alpha,R') \right\}, \text{ and}$$

$$M_*(s,(T,D)) = \operatorname*{argmin}_{m\in\mathcal{M}}{}^{\text{lex}} \left\{ \left(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\right) \,:\, m = ([s],\alpha,R') \right\}.$$

Optimality equations Opt$(\widehat{\Gamma})$, Opt$_{\text{Max}}(\widehat{\Gamma})$, Opt$_{\text{Min}}(\widehat{\Gamma})$, Opt$_{\geq}(\widehat{\Gamma})$ and Opt$_{\leq}(\widehat{\Gamma})$. Let $T : \mathcal{R} \to [S \to \mathbb{R}]$ and $D : \mathcal{R} \to [S \to \mathbb{N}]$. We write $(T, D) \models \text{Opt}_{\text{Max}}(\widehat{\Gamma})$ or $(T, D) \models \text{Opt}_{\text{Min}}(\widehat{\Gamma})$, respectively, if for all $s \in F$, we have $\big(\widetilde{T}(s), \widetilde{D}(s)\big) = (0, 0)$, and for all $s \in S \setminus F$, we have, respectively:

$$\big(\widetilde{T}(s), \widetilde{D}(s)\big) = \max_{m\in\mathcal{M}}{}^{\text{lex}} \left\{ \left(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\right) \,:\, m = ([s],\alpha,R') \right\}, \text{ or}$$

$$\big(\widetilde{T}(s), \widetilde{D}(s)\big) = \min_{m\in\mathcal{M}}{}^{\text{lex}} \left\{ \left(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\right) \,:\, m = ([s],\alpha,R') \right\}.$$

If $\widehat{\Gamma}$ is 0-player then Opt$_{\text{Max}}(\widehat{\Gamma})$ and Opt$_{\text{Min}}(\widehat{\Gamma})$ are equivalent to each other and denoted by Opt$(\widehat{\Gamma})$.

We write $(T, D) \models \text{Opt}_{\geq}(\widehat{\Gamma})$ or $(T, D) \models \text{Opt}_{\leq}(\widehat{\Gamma})$, respectively, if for all $s \in F$, we have $\big(\widetilde{T}(s), \widetilde{D}(s)\big) \geq^{\text{lex}} (0, 0)$ or $\big(\widetilde{T}(s), \widetilde{D}(s)\big) \leq^{\text{lex}} (0, 0)$, respectively; and for all $s \in S \setminus F$, we have, respectively:

$$\big(\widetilde{T}(s), \widetilde{D}(s)\big) \geq^{\text{lex}} \max_{m\in\mathcal{M}}{}^{\text{lex}} \left\{ \left(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\right) \,:\, m = ([s],\alpha,R') \right\}, \text{ or}$$

$$\big(\widetilde{T}(s), \widetilde{D}(s)\big) \leq^{\text{lex}} \min_{m\in\mathcal{M}}{}^{\text{lex}} \left\{ \left(T(R')^{\oplus}_{\alpha}(s), D(R')^{\boxplus}_{\alpha}(s)\right) \,:\, m = ([s],\alpha,R') \right\}.$$

The following Propositions 1 and 2 are simple but key properties of optimality equations and simple functions, which allow us to establish that if (T, D) is a solution of

0-player, 1-player, or 2-player optimality equations for $\widehat{\Gamma}$, then T is regionally simple and D is regionally constant.

Proposition 1. *Let $\alpha \in \mathcal{A}$; $R, R' \in \mathcal{R}$; and $R \xrightarrow{\alpha} R'$. If $F : R' \to \mathbb{R}$ is simple, then $F^{\oplus}_{\alpha} : R \to \mathbb{R}$ is simple.*

The following lemma can be proved by induction on the value of the function D, using Proposition 1.

Lemma 2 (Solution of Opt($\widehat{\Gamma}$) is regionally simple). *Let $\widehat{\Gamma}$ be a 0-player timed region graph. If $(T, D) \models Opt(\widehat{\Gamma})$ then T is regionally simple and D is regionally constant.*

4.1 Solving 1-Player Reachability-Time Optimality Equations $\mathrm{Opt}_{\mathrm{Max}}(\widehat{\Gamma})$

In this subsection we give a strategy improvement algorithm for solving maximum reachability-time optimality equations $\mathrm{Opt}_{\mathrm{Max}}(\widehat{\Gamma})$ for a 1-player timed region graph $\widehat{\Gamma}$.

Let Choose : $2^{\mathcal{M}} \to \mathcal{M}$ be an arbitrary function such that for every non-empty set of moves $M \subseteq \mathcal{M}$, we have Choose$(M) \in M$. We define the following strategy improvement operator $\mathrm{Improve}_{\mathrm{Max}}$:

$$\mathrm{Improve}_{\mathrm{Max}}(\chi, (T, D))(s) = \begin{cases} \chi(s) & \text{if } \chi(s) \in M^*(s, (T, D)), \\ \mathrm{Choose}(M^*(s, T)) & \text{if } \chi(s) \notin M^*(s, (T, D)). \end{cases}$$

For $F, F' : X \to \mathbb{R}$, we define functions $\max(F, F'), \min(F, F') : X \to \mathbb{R}$ by $\max(F, F')(s) = \max\{ F(s), F'(s) \}$ and $\min(F, F')(s) = \min\{ F(s), F'(s) \}$, for every $s \in X$. The following closure of simple functions under minimum and maximum operations, together with Proposition 1, yields the important Lemma 3.

Proposition 2. *Let $F, F' : R \to \mathbb{R}$ be simple functions defined on a region $R \in \mathcal{R}$. Then either $\min(F, F') = F$ and $\max(F, F') = F'$, or $\min(F, F') = F'$ and $\max(F, F') = F$. In particular, $\min(F, F')$ and $\max(F, F')$ are simple functions.*

Lemma 3 (Improvement preserves regional constancy of strategies). *If $\chi \in \Delta_{Max}$ is regionally constant, $T : \mathcal{R} \to [S \to \mathbb{R}]$ is regionally simple, and $D : \mathcal{R} \to [S \to \mathbb{N}]$ is regionally constant, then $\mathrm{Improve}_{Max}(\chi, (T, D))$ is regionally constant.*

Algorithm 1. Strategy improvement algorithm for $\mathrm{Opt}_{\mathrm{Max}}(\widehat{\Gamma})$.

1. (Initialisation) Choose a regionally constant positional strategy χ_0 for player Max in $\widehat{\Gamma}$; set $i := 0$.
2. (Value computation) Compute the solution (T_i, D_i) of $Opt(\widehat{\Gamma}\restriction\chi_i)$.
3. (Strategy improvement) If $\mathrm{Improve}_{\mathrm{Max}}(\chi_i, (T_i, D_i)) = \chi_i$, then return (T_i, D_i). Otherwise, set $\chi_{i+1} := \mathrm{Improve}_{\mathrm{Max}}(\chi_i, (T_i, D_i))$; set $i := i + 1$; and goto step 2.

Proposition 3 (Solutions of $\mathrm{Opt}_{\mathrm{Max}}(\widehat{\Gamma})$ from fixpoints of $\mathrm{Improve}_{\mathrm{Max}}$). *Let $\chi \in \Delta_{Max}$ and let $(T^\chi, D^\chi) \models Opt(\widehat{\Gamma}\restriction\chi)$. If $\mathrm{Improve}_{Max}(\chi, (T^\chi, D^\chi)) = \chi$ then we have $(T^\chi, D^\chi) \models \mathrm{Opt}_{Max}(\widehat{\Gamma})$.*

If $F, F', G, G' : \mathcal{R} \to [S \to \mathbb{R}]$ then we write $F \leq F'$ or $(F, G) \leq^{\text{lex}} (F', G')$, if for all $R \in \mathcal{R}$ and $s \in \overline{R}$, we have $F(R)(s) \leq F'(R)(s)$ or $\big(F(R)(s), G(R)(s)\big) \leq^{\text{lex}} \big(F'(R)(s), G'(R)(s)\big)$, respectively. Moreover, $F < F'$ or $(F, G) <^{\text{lex}} (F', G')$, if $F \leq F'$ or $(F, G) \leq^{\text{lex}} (F', G')$, and there is $R \in \mathcal{R}$ and $s \in R$, such that $F(R)(s) < F'(R)(s)$ or $\big(F(R)(s), G(R)(s)\big) <^{\text{lex}} \big(F'(R)(s), G'(R)(s)\big)$, respectively.

The following characterization of the solution of $\text{Opt}(\widehat{\Gamma})$ as the *maximum* solution of $\text{Opt}_{\leq}(\widehat{\Gamma})$ yields Lemma 4 which is key for termination (see the proof of Lemma 6).

Proposition 4 (Solution of Opt$(\widehat{\Gamma})$ dominates solutions of Opt$_{\leq}(\widehat{\Gamma})$). *If $(T, D) \models$ Opt$(\widehat{\Gamma})$ and $(T_{\leq}, D_{\leq}) \models$ Opt$_{\leq}(\widehat{\Gamma})$, then we have $(T_{\leq}, D_{\leq}) \leq^{lex} (T, D)$, and if $(T_{\leq}, D_{\leq}) \not\models$ Opt$(\widehat{\Gamma})$ then $(T_{\leq}, D_{\leq}) <^{lex} (T, D)$.*

Lemma 4 (Strict strategy improvement for Max). *Let $\chi, \chi' \in \Delta_{Max}$, let $(T, D) \models$ Opt$_{Min}(\widehat{\Gamma} \restriction \chi)$ and $(T', D') \models$ Opt$_{Min}(\widehat{\Gamma} \restriction \chi')$, and let $\chi' =$ Improve$_{Max}(\chi, (T, D))$. Then $(T, D) \leq^{lex} (T', D')$ and if $\chi \neq \chi'$ then $(T, D) <^{lex} (T', D')$.*

The following theorem is an immediate corollary of Lemmas 2 and 3 (the algorithm considers only regionally constant strategies), of Lemma 4 and finiteness of the number of regionally constant positional strategies for Max (the algorithm terminates), and of Proposition 3 (the algorithm returns a solution of optimality equations).

Theorem 3 (Correctness and termination of strategy improvement). *The strategy improvement algorithm terminates in finitely many steps and returns a solution (T, D) of Opt$_{Max}(\widehat{\Gamma})$, such that T is regionally simple and D is regionally constant.*

4.2 Solving 2-Player Reachability-Time Optimality Equations Opt$_{\text{MinMax}}(\widehat{\Gamma})$

In this subsection we give a strategy improvement algorithm for solving optimality equations $\text{Opt}_{\text{MinMax}}(\widehat{\Gamma})$ for a 2-player timed region graph $\widehat{\Gamma}$. The structure of the algorithm is very similar to that of Algorithm 1. The only difference is that in step 2. of every iteration we solve 1-player optimality equations $\text{Opt}_{\text{Max}}(\widehat{\Gamma} \restriction \mu)$ instead of 0-player optimality equations $\text{Opt}(\widehat{\Gamma} \restriction \chi)$. Note that we can perform step 2. of Algorithm 2 below by using Algorithm 1. Define the following strategy improvement operator $\text{Improve}_{\text{Min}}$:

$$\text{Improve}_{\text{Min}}(\mu, (T, D))(s) = \begin{cases} \mu(s) & \text{if } \mu(s) \in M_*(s, (T, D)), \\ \text{Choose}(M_*(s, (T, D))) & \text{if } \mu(s) \notin M_*(s, (T, D)). \end{cases}$$

Lemma 5 (Improvement preserves regional constancy of strategies). *If $\mu \in \Delta_{Min}$ is regionally constant, $T : \mathcal{R} \to [S \to \mathbb{R}]$ is regionally simple, and $D : \mathcal{R} \to [S \to \mathbb{R}]$ is regionally constant, then Improve$_{Min}(\mu, (T, D))$ is regionally constant.*

Algorithm 2. Strategy improvement algorithm for solving Opt$_{\text{MinMax}}(\widehat{\Gamma})$.

1. (Initialisation) Choose a regionally constant positional strategy μ_0 for player Min in $\widehat{\Gamma}$; set $i := 0$.
2. (Value computation) Compute the solution (T_i, D_i) of $Opt_{\text{Max}}(\widehat{\Gamma} \restriction \mu_i)$.

3. (Strategy improvement) If $\text{Improve}_{\text{Min}}(\mu_i, (T_i, D_i)) = \mu_i$, then return (T_i, D_i). Otherwise, set $\mu_{i+1} := \text{Improve}_{\text{Min}}(\mu_i, (T_i, D_i))$; set $i := i + 1$; and goto step 2.

Proposition 5 (Fixpoints of $\textbf{Improve}_{\textbf{Min}}$ are solutions of $\textbf{Opt}_{\textbf{MinMax}}(\widehat{\Gamma})$). *Let $\mu \in \Delta_{Min}$ and $(T^\mu, D^\mu) \models Opt_{Max}(\widehat{\Gamma}{\restriction}\mu)$. If $Improve_{Min}(\mu, (T^\mu, D^\mu)) = \mu$ then we have $(T^\mu, D^\mu) \models Opt_{MinMax}(\widehat{\Gamma})$.*

Proposition 6 (Solution of $\textbf{Opt}_{\textbf{Max}}(\widehat{\Gamma})$ is dominated by solutions of $\textbf{Opt}_{\geq}(\widehat{\Gamma})$). *If $(T, D) \models Opt_{Max}(\widehat{\Gamma})$ and $(T_\geq, D_\geq) \models Opt_\geq(\widehat{\Gamma})$, then we have $(T_\geq, D_\geq) \geq^{lex} (T, D)$, and if $(T_\geq, D_\geq) \not\models Opt_{Max}(\widehat{\Gamma})$ then $(T_\geq, D_\geq) >^{lex} (T, D)$.*

Lemma 6 (Strict strategy improvement for Min). *Let $\mu, \mu' \in \Delta_{Min}$, let $(T, D) \models Opt_{Max}(\widehat{\Gamma}{\restriction}\mu)$ and $(T', D') \models Opt_{Max}(\widehat{\Gamma}{\restriction}\mu')$, and let $\mu' = Improve_{Min}(\mu, (T, D))$. Then $(T, D) \geq^{lex} (T', D')$ and if $\mu \neq \mu'$ then $(T, D) >^{lex} (T', D')$.*

Proof. First we argue that $(T, D) \models \text{Opt}_\geq(\widehat{\Gamma}{\restriction}\mu')$ which by Proposition 6 implies that $(T, D) \geq^{\text{lex}} (T', D')$. Indeed for every $s \in S \setminus F$, if $\mu(s) = ([s], \alpha, R)$ and $\mu'(s) = ([s], \alpha', R')$ then we have

$$\big(\widetilde{T}(s), \widetilde{D}(s)\big) = \big(T(R)^{\oplus}_{\alpha}(s), D(R)^{\boxplus}_{\alpha}(s)\big) \geq^{\text{lex}} \big(T(R')^{\oplus}_{\alpha'}(s), D(R')^{\boxplus}_{\alpha'}(s)\big),$$

where the equality follows from $(T, D) \models \text{Opt}_{\text{Max}}(\widehat{\Gamma}{\restriction}\mu)$, and the inequality follows from the definition of $\text{Improve}_{\text{Min}}$. Moreover, if $\mu \neq \mu'$ then there is $s \in S_{\text{Min}} \setminus F$ for which the above inequality is strict. Then $(T, D) \not\models \text{Opt}_{\text{Max}}(\widehat{\Gamma}{\restriction}\mu')$ because every vertex $R \in \mathcal{R}_{\text{Min}}$ in $\widehat{\Gamma}{\restriction}\mu'$ has a unique successor, and hence again by Proposition 6 we conclude that $(T, D) >^{\text{lex}} (T', D')$. □

The following theorem is an immediate corollary of Theorem 3 and Lemma 5, of Lemma 6 and finiteness of the number of regionally constant positional strategies for Min, and of Proposition 5.

Theorem 4 (Correctness and termination of strategy improvement). *The strategy improvement algorithm terminates in finitely many steps and returns a solution (T, D) of $Opt_{MinMax}(\widehat{\Gamma})$, such that T is regionally simple and D is regionally constant.*

5 Complexity

Lemma 7 (Complexity of strategy improvement). *Let $\widehat{\Gamma_0}$, $\widehat{\Gamma_1}$, and $\widehat{\Gamma_2}$ be 0-player, 1-player, and 2-player timed region graphs, respectively. A solution of $Opt(\widehat{\Gamma_0})$ can be computed in time $O(|\mathcal{R}|)$. The strategy improvement algorithms for $Opt_{Max}(\widehat{\Gamma_1})$ and $Opt_{MinMax}(\widehat{\Gamma_2})$ terminate in $O(|\mathcal{R}|)$ iterations.*

Since the number $|\mathcal{R}|$ of regions is at most exponential in the size of a timed automaton [3], it follows that the strategy improvement algorithm runs in exponential time, and hence solving reachability-time games is in EXPTIME. The reachability problem for timed automata with three clocks is PSPACE-complete [13]. We show that solving 2-player reachability games on timed automata with two clocks is EXPTIME-complete. We use a reduction from countdown games [16] for EXPTIME-hardness.

Theorem 5 (Complexity of reachability(-time) games on timed automata). *Problems of solving reachability and reachability-time games are EXPTIME-complete on timed automata with at least two clocks.*

References

1. Y. Abdeddaïm, E. Asarin, and O. Maler. Scheduling with timed automata. *Theor. Comput. Sci.*, 354(2):272–300, 2006.
2. R. Alur, M. Bernadsky, and P. Madhusudan. Optimal reachability for weighted timed games. In *ICALP 2004*, volume 3142 of *LNCS*, pages 122–133. Springer, 2004.
3. R. Alur and D. L. Dill. A theory of timed automata. *Theor. Comput. Sci.*, 126(2):183–235, 1994.
4. E. Asarin and O. Maler. As soon as possible: Time optimal control for timed automata. In *HSCC'99*, volume 1569 of *LNCS*, pages 19–30. Springer, 1999.
5. E. Asarin, O. Maler, A. Pnueli, and J. Sifakis. Controller synthesis for timed automata. In *IFAC Symp. on System Structure and Control*, pages 469–474. Elsevier, 1998.
6. G. Behrmann, A. Fehnker, T. Hune, K. G. Larsen, P. Pettersson, J. Romijn, and F. W. Vaandrager. Minimum-cost reachability for priced timed automata. In *HSCC 2001*, volume 2034 of *LNCS*, pages 147–161. Springer, 2001.
7. P. Bouyer. Weighted timed automata: model-checking and games. *Electr. Notes Theor. Comput. Sci.*, 158:3–17, 2006.
8. P. Bouyer, T. Brihaye, V. Bruyère, and J.-F. Raskin. On the optimal reachability problem on weighted timed automata. *Form. Method. Syst. Des.*, 2007. To appear.
9. P. Bouyer, T. Brihaye, and N. Markey. Improved undecidability results on weighted timed automata. *Information Processing Letters*, 98(5):188–194, 2006.
10. P. Bouyer, F. Cassez, E. Fleury, and K. G. Larsen. Optimal strategies in priced timed game automata. In *FSTTCS 2004*, volume 3328 of *LNCS*, pages 148–160. Springer, 2004.
11. P. Bouyer, K. G. Larsen, N. Markey, and J. I. Rasmussen. Almost optimal strategies in one-clock priced timed automata. In *FSTTCS'06*, volume 4337 of *LNCS*, pages 345–356. Springer, 2006.
12. T. Brihaye, T. A. Henzinger, V. S. Prabhu, and J.-F. Raskin. Minimum-time reachability in timed games. In *ICALP 2007*, volume 4596 of *LNCS*. Springer, 2007.
13. C. Courcoubetis and M. Yannakakis. Minimum and maximum delay problems in real-time systems. *Form. Method. Syst. Des.*, 1:385–415, 1992.
14. L. de Alfaro, M. Faella, T. A. Henzinger, R. Majumdar, and M. Stoelinga. The element of surprise in timed games. In *CONCUR 2003*, volume 2761 of *LNCS*, pages 144–158, 2003.
15. T. A. Henzinger and V. S. Prabhu. Timed alternating-time temporal logic. In *FORMATS 2006*, volume 4202 of *LNCS*, pages 1–17. Springer, 2006.
16. M. Jurdziński, F. Laroussinie, and J. Sproston. Model checking probabilistic timed automata with one or two clocks. In *TACAS 2007*, volume 4424 of *LNCS*, pages 170–184, 2007.
17. M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 1994.
18. J. I. Rasmussen, K. G. Larsen, and K. Subramani. On using priced timed automata to achieve optimal scheduling. *Form. Method. Syst. Des.*, 29:97–114, 2006.
19. J. Vöge and M. Jurdziński. A discrete strategy improvement algorithm for solving parity games. In *CAV 2000*, volume 1855 of *LNCS*, pages 202–215. Springer, 2000.