# On decision problems for timed automata

Olivier Finkel
*Equipe de Logique Mathématique*,
U.F.R. de Mathématiques, Université Paris 7
*2 Place Jussieu 75251 Paris cedex 05, France.*
finkel@logique.jussieu.fr

**Abstract**

We solve some decision problems for timed automata which were raised by Tripakis in [9] and by Asarin in [3]. In particular, we show that one cannot decide whether a given timed automaton is determinizable or whether the complement of a timed regular language is timed regular.

**Keywords:** Timed automata; timed regular languages; decision problems; determinizability; complementability; universality problem.

## 1 Introduction

We assume the reader to be familiar with the basic theory of timed languages and timed automata (TA) [1].
The set of positive reals will be denoted $\mathcal{R}$. A (finite length) timed word over a finite alphabet $\Sigma$ is in the form $t_1.a_1.t_2.a_2 \ldots t_n.a_n$, where, for all integers $i \in [1, n]$, $t_i \in \mathcal{R}$ and $a_i \in \Sigma$. It may be seen as a *time-event sequence*, where the $t_i \in \mathcal{R}$ represent time lapses between events and the letters $a_i \in \Sigma$ represent events. The set of all (finite length) timed words over a finite alphabet $\Sigma$ is the set $(\mathcal{R}\times\Sigma)^\star$. A timed language is a subset of $(\mathcal{R}\times\Sigma)^\star$. The complement ( in $(\mathcal{R}\times\Sigma)^\star$ ) of a timed language $L \subseteq (\mathcal{R}\times\Sigma)^\star$ is $(\mathcal{R}\times\Sigma)^\star - L$ denoted $L^c$.
We consider a basic model of timed automaton, as introduced in [1]. A timed automaton $\mathcal{A}$ has a finite set of states and a finite set of transitions. Each transition is labelled with a letter of a finite input alphabet $\Sigma$. We assume that each transition of $\mathcal{A}$ has a set of clocks to reset to zero and only *diagonal-free* clock guard [1]. As usual, we denote $L(\mathcal{A})$ the timed language accepted (by final states) by the timed automaton $\mathcal{A}$. A timed language $L \subseteq (\mathcal{R} \times \Sigma)^\star$ is said to be timed regular iff there is a timed automaton $\mathcal{A}$ such that $L = L(\mathcal{A})$.

Many decision problems for timed automata have been studied and partially solved, see [2] for a survey of these results. Some decision problems were recently raised by Tripakis in [9] and by Asarin in [3]. We give in this paper the answer to several questions of [9, 3]. In particular, we show that one cannot decide whether a given timed automaton is determinizable or whether the complement of a timed regular language is timed regular. For that purpose we use a method which is very similar to that one used in [4] to prove undecidability results about infinitary rational relations.

# 2 Complementability and determinizability

We first state our main result about the undecidability of determinizability or regular complementability for timed regular languages.

**Theorem 2.1.** *It is undecidable to determine, for a given TA $\mathcal{A}$, whether*

1. *$L(\mathcal{A})$ is accepted by a deterministic TA.*

2. *$L(\mathcal{A})^c$ is accepted by a TA.*

**Proof.** It is well known that the class of timed regular languages is not closed under complementation. Let $\Sigma$ be a finite alphabet and let $a \in \Sigma$. Let $A$ be the set of timed words in the form $t_1.a.t_2.a \ldots t_n.a$, where, for all integers $i \in [1, n]$, $t_i \in \mathcal{R}$ and there is a pair of integers $(i, j)$ such that $i, j \in [1, n]$, $i < j$, and $t_{i+1} + t_{i+2} + \ldots + t_j = 1$. The timed language $A$ is formed by timed words containing only letters $a$ and such that there is a pair of $a$'s which are separated by a time distance 1. The timed language $A$ is regular but its complement can not be accepted by any timed automaton because otherwise this timed automaton should have an unbounded number of clocks to check that no pair of $a$'s is separated by a time distance 1, [1].

We shall use the undecidability of the universality problem for timed regular languages: one cannot decide, for a given timed automaton $\mathcal{A}$ with input alphabet $\Sigma$, whether $L(\mathcal{A}) = (\mathcal{R} \times \Sigma)^\star$.

Let $c$ be an additional letter not in $\Sigma$. For a given timed regular language $L \subseteq (\mathcal{R} \times \Sigma)^\star$, we are going to construct another timed language $\mathcal{L}$ over the alphabet $\Gamma = \Sigma \cup \{c\}$ defined as the union of the following three languages.

- $\mathcal{L}_1 = L.(\mathcal{R} \times \{c\}).(\mathcal{R} \times \Sigma)^\star$
- $\mathcal{L}_2$ is the set of timed words over $\Gamma$ having not any letters $c$ or having at least two letters $c$.
- $\mathcal{L}_3 = (\mathcal{R} \times \Sigma)^\star.(\mathcal{R} \times \{c\}).A$, where $A$ is the above defined timed regular language over the alphabet $\Sigma$.

The timed language $\mathcal{L}$ is regular because $L$ and $A$ are regular timed languages. There are now two cases.

(1) **First case.** $L = (\mathcal{R}\times\Sigma)^\star$. Then $\mathcal{L} = (\mathcal{R}\times(\Sigma\cup\{c\}))^\star$. Therefore $\mathcal{L}$ has the minimum possible complexity. $\mathcal{L}$ is of course accepted by a deterministic timed automaton (without any clock). Moreover its complement $\mathcal{L}^c$ is empty thus it is also accepted by a deterministic timed automaton (without any clock).

(2) **Second case.** $L$ is strictly included into $(\mathcal{R}\times\Sigma)^\star$. Then there is a timed word $u = t_1.a_1.t_2.a_2\ldots t_n.a_n \in (\mathcal{R}\times\Sigma)^\star$ which does not belong to $L$. Consider now a timed word $x \in (\mathcal{R}\times\Sigma)^\star$. It holds that $u.1.c.x \in \mathcal{L}$ iff $x \in A$. Then we have also : $u.1.c.x \in \mathcal{L}^c$ iff $x \in A^c$.
We are going to show that $\mathcal{L}^c$ is not timed regular. Assume on the contrary that there is a timed automaton $\mathcal{A}$ such that $\mathcal{L}^c = L(\mathcal{A})$. There are only finitely many possible global states (including the clock values) of $\mathcal{A}$ after the reading of the initial segment $u.1.c$. It is clearly not possible that the timed automaton $\mathcal{A}$, from these global states, accept all timed words in $A^c$ and only these ones, for the same reasons which imply that $A^c$ is not timed regular. Thus $\mathcal{L}^c$ is not timed regular. This implies that $\mathcal{L}$ is not accepted by any deterministic timed automaton because the class of deterministic regular timed languages is closed under complement.

In the first case $\mathcal{L}$ is accepted by a deterministic timed automaton and $\mathcal{L}^c$ is timed regular. In the second case $\mathcal{L}$ is not accepted by any deterministic timed automaton and $\mathcal{L}^c$ is not timed regular. But one cannot decide which case holds because of the undecidability of the universality problem for timed regular languages. □

Below $TA(n,K)$ denotes the class of timed automata having at most $n$ clocks and where constants are at most $K$. In [9], Tripakis stated the following problems which are similar to the above ones but with “bounded resources".

Problem 10 of [9]. Given a TA $\mathcal{A}$ and non-negative integers $n, K$, does there exist a TA $\mathcal{B} \in TA(n,K)$ such that $L(\mathcal{B})^c = L(\mathcal{A})$ ? If so, construct such a $\mathcal{B}$.

Problem 11 of [9]. Given a TA $\mathcal{A}$ and non-negative integers $n, K$, does there exist a deterministic TA $\mathcal{B} \in TA(n,K)$ such that $L(\mathcal{B}) = L(\mathcal{A})$ ? If so, construct such a $\mathcal{B}$.

Tripakis showed that these problems are not algorithmically solvable. He asked also whether these bounded-resource versions of previous problems remain undecidable if we do not require the construction of the witness $\mathcal{B}$, i.e. if we omit the sentence “If so construct such a $\mathcal{B}$" in the statement of Problems 10 and 11.
It is easy to see, from the proof of preceding Theorem, that this is actually the case because we have seen that, in the first case, $\mathcal{L}$ and $\mathcal{L}^c$ are accepted by deterministic timed automata *without any clock*.

# 3 Minimization of the number of clocks

The following problem was shown to be undecidable by Tripakis in [9].

Problem 5 of [9]. Given a TA $\mathcal{A}$ with $n$ clocks, does there exists a TA $\mathcal{B}$ with $n-1$ clocks, such that $L(\mathcal{B}) = L(\mathcal{A})$ ? If so, construct such a $\mathcal{B}$.

The corresponding decision problem, where we require only a Yes / No answer but no witness in the case of a positive answer, was left open in [9].
Using a very similar reasoning as in the preceding section, we can prove that this problem is also undecidable.

**Theorem 3.1.** *Let $n \geq 2$ be a positive integer. It is undecidable to determine, for a given TA $\mathcal{A}$ with n clocks, whether there exists a TA $\mathcal{B}$ with $n-1$ clocks, such that $L(\mathcal{B}) = L(\mathcal{A})$*

**Proof.** Let $\Sigma$ be a finite alphabet and let $a \in \Sigma$. Let $n \geq 2$ be a positive integer, and $A_n$ be the set of timed words in the form $t_1.a.t_2.a \ldots t_k.a$, where, for all integers $i \in [1, k]$, $t_i \in \mathcal{R}$ and there are $n$ pairs of integers $(i, j)$ such that $i, j \in [1, k]$, $i < j$, and $t_{i+1} + t_{i+2} + \ldots + t_j = 1$. The timed language $A_n$ is formed by timed words containing only letters $a$ and such that there are $n$ pairs of $a$'s which are separated by a time distance 1. $A_n$ is a timed regular language but it can not be accepted by any timed automaton with less than $n$ clocks.

Let $c$ be an additional letter not in $\Sigma$. For a given timed regular language $L \subseteq (\mathcal{R} \times \Sigma)^\star$, we construct another timed language $\mathcal{V}_n$ over the alphabet $\Gamma = \Sigma \cup \{c\}$ defined as the union of the following three languages.

- $\mathcal{V}_{n,1} = L.(\mathcal{R} \times \{c\}).(\mathcal{R} \times \Sigma)^\star$
- $\mathcal{V}_{n,2}$ is the set of timed words over $\Gamma$ having not any letters $c$ or having at least two letters $c$.
- $\mathcal{V}_{n,3} = (\mathcal{R} \times \Sigma)^\star.(\mathcal{R} \times \{c\}).A_n$.

The timed language $\mathcal{V}_n$ is regular because $L$ and $A_n$ are regular timed languages. There are now two cases.

(1) **First case.** $L = (\mathcal{R} \times \Sigma)^\star$. Then $\mathcal{V}_n = (\mathcal{R} \times (\Sigma \cup \{c\}))^\star$, thus $\mathcal{V}_n$ is accepted by a (deterministic) timed automaton *without any clock*.

(2) **Second case.** $L$ is strictly included into $(\mathcal{R} \times \Sigma)^\star$. Then there is a timed word $u = t_1.a_1.t_2.a_2 \ldots t_k.a_k \in (\mathcal{R} \times \Sigma)^\star$ which does not belong to $L$. Consider now a timed word $x \in (\mathcal{R} \times \Sigma)^\star$. It holds that $u.1.c.x \in \mathcal{V}_n$ iff $x \in A_n$.
Towards a contradiction, assume that $\mathcal{V}_n$ is accepted by a timed automaton $\mathcal{B}$ with at most $n-1$ clocks. There are only finitely many possible global states (including the clock values) of $\mathcal{B}$ after the reading of the initial segment $u.1.c$. It is clearly not possible that the timed automaton $\mathcal{B}$, from these global states, accept all timed words in $A_n$ and only these ones, because it has less than $n$ clocks.

But one cannot decide which case holds because of the undecidability of the universality problem for timed regular languages accepted by timed automata with $n$ clocks, where $n \geq 2$. □

**Remark 3.2.** *For timed automata with only one clock, the inclusion problem, hence also the universality problem, have recently been shown to be decidable by Ouaknine and Worrell [8]. Then the above method can not be applied. It is easy to see that it is decidable whether a timed regular language accepted by a timed automaton with only one clock is also accepted by a timed automaton without any clock.*

# 4 Concluding remarks

We have restricted here the study to the case of *finite* timed words as in [9, 3]. However the above results can be easily extended to the case of timed regular $\omega$-languages accepted by Büchi timed automata.

The simple idea behind the proofs was already used in [4] and relies heavily on the undecidability of the universality problem.

It could be easily used in other contexts, for instance to study the notion of ambiguity for context-free languages. Ginsburg and Ullian proved in [5] that one cannot decide whether a given context-free language is non-ambiguous or inherently ambiguous. We know that the class of inherently ambiguous context-free languages can be partitioned in an infinite hierarchy by considering the degree of ambiguity of a context-free language [6]. Moreover in recent works of Wich and Naji the context-free languages which are inherently ambiguous of infinite degrees can also be distinguished by considering the growth-rate of their ambiguity with respect to the length of the words [7, 10]. We are not aware of published results about the decidability of membership to subclasses of context-free languages defined with these notions of degrees of ambiguity.

Using the undecidability of the universality problem for context-free languages and a similar method as in this paper, we can easily prove results like: one cannot decide whether a given context-free language has a degree of ambiguity which is smaller than $k$, where $k \geq 2$ is a positive integer, or which is smaller than "exponentially ambiguous" (in the sense of Naji and Wich).

# References


[1] R. Alur and D. Dill, A Theory of Timed Automata, Theoretical Computer Science, Volume 126, p. 183-235, 1994.

[2] R. Alur and P. Madhusudan, Decision Problems for Timed Automata: A Survey, in Formal Methods for the Design of Real-Time Systems, International School on Formal Methods for the Design of Computer, Communication and Software Systems, SFM-RT 2004, Revised Lectures. Lecture Notes in Computer Science, Volume 3185, Springer, 2004, p. 1-24.

[3] E. Asarin, Challenges in Timed Languages, From Applied Theory to Basic Theory, Bulletin of the European Association for Theoretical Computer Science, Volume 83, p. 106-120, June 2004.

[4] O. Finkel, Undecidability of Topological and Arithmetical Properties of Infinitary Rational Relations, RAIRO-Theoretical Informatics and Applications, Volume 37 (2), 2003, p. 115-126.

[5] S. Ginsburg and J.S. Ulllian, Ambiguity in Context Free Languages, JACM 13 (1), 1966, p. 62-89.

[6] H. A. Maurer, The Existence of Context Free Languages which are Inherently Ambiguous of any Degree, Dept. of Mathematics, Research Series, University of Calgary, 1968.

[7] M. Naji, Grad der Mehrdeutigkeit Kontextfreier Grammatiken und Sprachen, Diplomarbeit, FB Informatik, Johann-Wolfgang-Goethe-Universität, Frankfurt am Main, 1998.

[8] J. Ouaknine and J. Worrell, On the Language Inclusion Problem for Timed Automata: Closing a Decidability Gap, in the Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science , LICS 2004, Turku, Finland, IEEE Computer Society, 2004, p. 54-63.

[9] S. Tripakis, Folk Theorems on the Determinization and Minimization of Timed Automata, in the Proceedings of FORMATS'2003, Lecture Notes in Computer Science, Volume 2791, p. 182-188, 2004.

[10] K. Wich, Exponential Ambiguity of Context-free Grammars, Proc. of 4th Int. Conf. on Developments in Language Theory 1999, World Scientific, Singapore.