Fundamenta Informaticae XXI (2001) 1001–1025
IOS Press

Undecidability Results for Timed Automata with Silent Transitions*

Patricia Bouyer†
LSV, ENS Cachan, CNRS, France
bouyer@lsv.ens-cachan.fr

Serge Haddad
LSV, ENS Cachan, CNRS, France
haddad@lsv.ens-cachan.fr

Pierre-Alain Reynier‡
LIF, Université de Provence, CNRS, France
pierre-alain.reynier@lif.univ-mrs.fr

Abstract. In this work, we study decision problems related to timed automata with silent transitions (TA_ε) which strictly extend the expressiveness of timed automata (TA). We first answer negatively a central question raised by the introduction of silent transitions: can we decide whether the language recognized by a TA_ε can be recognized by some TA? Then we establish in the framework of TA_ε some old open conjectures that O. Finkel has recently solved for TA. His proofs follow a generic scheme which relies on the fact that only a finite number of configurations can be reached by a TA while reading a timed word. This property does not hold for TA_ε, the proofs in the framework of TA_ε thus require more elaborated arguments. We establish undecidability of complementability, minimization of the number of clocks, and closure under shuffle. We also show these results in the framework of infinite timed languages.

Keywords: Timed automata, silent transitions, decidability.

Address for correspondence: LSV, ENS Cachan
61, avenue du Président Wilson
94230 Cachan
France

*Work partly supported by project DOTS (ANR-06-SETI-003).
†Partly supported by a Marie Curie fellowship from the EU Commission.
‡Partly supported by a Lavoisier Postdoctoral fellowship from the Egide Foundation (French Foreign Office).

1. Introduction

The model of timed automata has been proposed by Alur and Dill in the early 90's as a model for real-time systems [3, 4]. A timed automaton is a finite automaton which can manipulate real-valued variables called clocks, that evolve synchronously with time, can be tested, and can be reset to zero. One of the fundamental properties of this model is that reachability properties can be decided, even though the set of configurations of a timed automaton is in general infinite. Since then, this model has attracted much attention, as it is very appropriate for verification purposes.

A constant interest goes to the theoretical understanding of this model, and to the theoretical foundations of timed languages. Indeed, the classical (untimed) regular formal languages enjoy very nice properties, like the equivalence of first-order logic with aperiodic regular languages, whose robustness cannot be denied.

The case of timed languages is much less satisfactory, as they do not enjoy those nice logical and algebraic characterizations, though this subject has inspired several approaches [24, 14, 17, 15, 8, 11, 12, 21, 13]. *The* right class of timed language has probably not yet been investigated, and much work is still required to really understand and formalize the theoretical foundations of timed languages [7].

Timed automata (and thus the set of timed regular languages) are neither closed under complementation nor determinizable. This makes it difficult to propose equivalent logical languages because the closure by negation is somewhat the quintessence of logics. Hence, either we need to forget about negation in the logics [24, 10], or we restrict to subclasses of languages closed by complementation [5, 17, 13], or we try to better understand the role of complementation. The paper [23] follows this idea, and asks questions like "Is a timed automaton complementable into another timed automaton?" or "Can a timed automaton be determinized?". The proof of Tripakis therein yields that those two problems are undecidable, as soon as we require that a witnessing automaton be constructed. He also provides such proofs requiring the construction of witnesses for various other problems like minimizing the number of clocks required to recognize a given timed language, *etc*. In [18], Finkel improved quite a lot the above-mentioned proofs by proving that all these problems are undecidable, even if we do not require the construction of witnessing automata.

Whatever the modelling framework, silent transitions are very useful: for instance they naturally occur in the design of modular systems where they can correspond to internal communications within a component, or it can be used as an abstraction device in order to compare an implementation with respect to its specification. In timed systems, they can furthermore be used to model discrete-time behaviours embedded in continuous environment. From the verification point of view, the standard symbolic analysis techniques (like the construction of the region automaton, or the construction of the zone-based simulation graph) apply to timed automata with silent transitions with no extra cost. Regarding expressiveness, the situation is different: Contrary to finite-state systems where silent transitions neither add expressiveness nor conciseness, it is well-known that silent transitions *do add* extra power to timed automata [9] and that on a specific timed input, the "branching behaviour" of a timed automaton is infinite with silent transitions and finite without.

In this paper, we address three significant problems related to the expressiveness of timed automata with silent transitions. The first question is: "Are silent transitions required in a specific timed model?". A purely reactive system should be modelled without silent transitions (corresponding somehow to proactive behaviours). The second question is then: "Which language operations do preserve the class of timed regular languages?". A modular design requires operations like union, complementation, etc. The last question is finally: "Which resources are really required in order to implement a system?". This topic takes even more sense in timed systems which are often embedded in some environment.

More precisely, we prove that it is not possible to:

Expressiveness

- decide whether an ε-timed regular language is timed regular (*i.e.*, if it is possible to remove silent transitions in timed automata), see Section 3;

Closure

- decide whether the complement of a ε-timed regular language is ε-timed regular, see Section 4;
- decide whether the shuffle[1] of two (ε-)timed regular languages is ε-timed regular, see Section 6.

Resources

- compute the minimal number of clocks needed to recognize an ε-timed regular language, see Section 5;

Finally, we extend all previous results, proved for finite timed words, to infinite timed words and to timed automata with a Büchi acceptance condition, see Section 7.

This paper builds up on previous works [9, 18]. We extend [9] with our result on expressiveness of silent transitions and we extend undecidability results of [18] to the framework of timed languages accepted by timed automata with silent transitions. Though we follow the same lines, the extension is far from trivial as results of [18] heavily rely on the the finitely branching property that timed automata *without* silent transitions enjoy.

2. Preliminaries

2.1. Timed words, timed languages

If S is a set, S^* denotes the set of all finite words over S whereas S^ω denotes the set of infinite words over S. We use classical notations like $\mathbb{R}_{\geq 0}$ or $\mathbb{Q}_{\geq 0}$ for the set of nonnegative real numbers (resp. nonnegative rational numbers).

Let Σ be a fixed finite alphabet. A finite (resp. infinite) *timed word* w over Σ is an element $w = (a_0, \tau_0)(a_1, \tau_1) \dots (a_n, \tau_n) \dots$ in $(\Sigma \times \mathbb{R}_{\geq 0})^*$ (resp. $(\Sigma \times \mathbb{R}_{\geq 0})^\omega$) such that for every $i \geq 0$, $a_i \in \Sigma$, $\tau_i \in \mathbb{R}_{\geq 0}$ and $\tau_{i+1} \geq \tau_i$. The value τ_k gives the absolute date at which action a_k occurs. Given $d \in \mathbb{R}_{\geq 0}$, we define the timed word $w + d = (a_0, \tau_0 + d)(a_1, \tau_1 + d) \dots (a_n, \tau_n + d) \dots$. We denote by $\mathcal{TW}^*(\Sigma)$ (resp. $\mathcal{TW}^\omega(\Sigma)$) the set of finite (resp. infinite) timed words over Σ. A *timed language over finite (resp. infinite) words* is a subset of $\mathcal{TW}^*(\Sigma)$ (resp. $\mathcal{TW}^\omega(\Sigma)$). Let $\mathcal{L}$ be a timed language, then $\overline{\mathcal{L}}$ denotes its complement. Let w be a timed word over Σ and $a \in \Sigma$, then $|w|_a$ is the number of occurrences of letter a in w. Finally, let us denote *Untimed* the operator which maps a timed word to the associated untimed word obtained by erasing the dates of actions.

[1]The shuffle operation corresponds to two tasks to be executed on the same processor by time sharing which have been extensively studied for monoprocessor scheduling problems.

2.2. Timed automata

Timed automata have been introduced in the 90's by Alur and Dill as a model for representing real-time systems [3, 4]. A timed automaton is a classical untimed finite automaton to which are associated a finite set of nonnegative real-valued variables called *clocks*.

Syntax. Let X be a finite set of clocks. We assume the time domain be the set $\mathbb{R}_{\geq 0}$ of nonnegative real numbers. A *valuation* v over X is a mapping $v : X \to \mathbb{R}_{\geq 0}$. Let $U \subseteq X$, the valuation $v[U \leftarrow 0]$ resets each clock of U to zero, *i.e.*, maps each clock $x \in U$ to 0, and each other clock $x \notin U$ to $v(x)$. Let $d \in \mathbb{R}_{\geq 0}$, the valuation $v + d$ maps every clock $x \in X$ to $v(x) + d$.

We write $\mathcal{C}(X)$ for the set of *(clock) constraints* over X consisting of conjunctions of atomic formulas of the form $x \bowtie h$ for $x \in X$, $h \in \mathbb{Q}_{\geq 0}$ is a nonnegative rational number, and $\bowtie \ \in \{<, \leq, =, \geq, >\}$. Such constraints are interpreted over valuations, and we write $v \models \gamma$ if valuation v satisfies the clock constraint γ. It is defined in a natural way by $v \models (x \bowtie h)$ whenever $v(x) \bowtie h$, and $v \models (\gamma_1 \wedge \gamma_2)$ whenever $v \models \gamma_1$ and $v \models \gamma_2$.

Definition 2.1. (Timed automaton)
Let Σ be a finite alphabet. A *timed automaton* over Σ is a tuple $\mathcal{A} = (L, \ell_0, X, E, F)$ where:

- L is a finite set of locations,
- $\ell_0 \in L$ is the initial location,
- X is a finite set of clocks,
- $E \subseteq L \times \mathcal{C}(X) \times \Sigma \times 2^X \times L$ is a finite set of edges, and
- F is the set of final locations.

An edge $e = \langle \ell, \gamma, a, U, \ell' \rangle \in E$ represents a transition from location ℓ to location ℓ' with label a, guard γ and reset U.

Let Σ be a finite alphabet, and let ε be a fresh symbol not in Σ. We write TA for the class of timed automata over Σ, and TA_ε for the set of timed automata over the alphabet $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$. The new event ε is a *silent action* and it is *unobservable*. A transition labelled by a silent action will be called a *silent transition*.

Let $\mathcal{A}$ be a TA or a TA_ε. The *granularity* of $\mathcal{A}$ is the smallest positive integer d such that each elementary constraint $x \bowtie h$ in $\mathcal{A}$ is such that $d \cdot h \in \mathbb{N}$. We define $\mathbb{N}_d = \{k/d \mid k \in \mathbb{N}\}$. We extend the notion of *modulo* w.r.t. $\mathbb{Q}_{>0}$. Let $r \in \mathbb{Q}_{>0}$, then we define $x \mod r = x - nr$ with $n = \max\{i \in \mathbb{Z} \mid ir \leq x\}$ and $x \equiv y \mod r$ iff $(x - y \mod r) = 0$.

If $\mathcal{A}$ is a TA, we say it is *deterministic* whenever given two distinct transitions $\langle \ell, \gamma_1, a, U_1, \ell'_1 \rangle$ and $\langle \ell, \gamma_2, a, U_2, \ell'_2 \rangle$, it holds that $\gamma_1 \wedge \gamma_2$ is not satisfiable.

Semantics. We give the semantics of a timed automaton as a timed transition system and then the corresponding accepted timed language. Let $\mathcal{A} = (L, \ell_0, X, E, F, R)$ be a TA over Σ (resp. TA_ε). It defines the timed transition system $S_{\mathcal{A}} = (Q, q_0, \to)$ where:

- $Q = L \times (\mathbb{R}_{\geq 0})^X$ is the set of states also called *configurations*,
- $q_0 = (\ell_0, \vec{0})$ is the initial state[2],
- and the transition relation $\rightarrow$ is composed of the following moves:
  - *delay moves:* $(\ell, v) \xrightarrow{d} (\ell, v + d)$ for every $d \in \mathbb{R}_{\geq 0}$;
  - *discrete moves:* $(\ell, v) \xrightarrow{a} (\ell', v')$ iff there exists some transition $e = (\ell, \gamma, a, U, \ell') \in E$ such that $v \models \gamma$, and $v' = v[U \leftarrow 0]$.

A *timed execution* of $\mathcal{A}$ is a (finite or infinite) path $\varrho : (\ell_0, v_0) \xrightarrow{d_0} (\ell_0, v_0 + d_0) \xrightarrow{a_0} (\ell_1, v_1) \xrightarrow{d_1} (\ell_1, v_1 + d_1) \xrightarrow{a_1} \ldots$ in $S_{\mathcal{A}}$ starting in the initial state q_0 (*i.e.*, $v_0 = \vec{0}$) and alternating between delay and discrete moves. Given such a timed execution, $\tau_i = \sum_{k \leq i} d_k$ denotes the absolute date at which transition labelled by a_i occurs. The *duration* of ϱ is the (eventually infinite) sum of all delays along ϱ, *i.e.* $\sup_i(\tau_i)$.

If τ is a value smaller than or equal to the duration of a finite execution ϱ, we write $(\ell^-_{\varrho,\tau}, v^-_{\varrho,\tau})$ for the first configuration along ϱ in which the automaton is at date τ, and $(\ell^+_{\varrho,\tau}, v^+_{\varrho,\tau})$ for the last configuration at date τ. More formally, defining $i^- = \max\{i \mid \tau_i < \tau\}$ and $i^+ = \max\{i \mid \tau_i \leq \tau\}$ with the convention $\max(\emptyset) = 0$, then:

$$\begin{cases} (\ell^-_{\varrho,\tau}, v^-_{\varrho,\tau}) = (\ell_{i^-}, v_{i^-} + \tau - \tau_{i^-}), \text{ and} \\ (\ell^+_{\varrho,\tau}, v^+_{\varrho,\tau}) = (\ell_{i^+}, v_{i^+} + \tau - \tau_{i^+}). \end{cases}$$

For instance, if a single transition occurs at time τ, $(\ell^-_{\varrho,\tau}, v^-_{\varrho,\tau})$ is the configuration before the transition is fired, whereas $(\ell^+_{\varrho,\tau}, v^+_{\varrho,\tau})$ is the configuration after the transition is fired. A major observation is that when $\tau > 0$ then $\forall x \in X, v^-_{\varrho,\tau}(x) > 0$.

Let ϱ be a timed execution in a TA $\mathcal{A}$. The label of ϱ is the timed word $w = (a_0, \tau_0)(a_1, \tau_1) \ldots$ (with τ_i defined as previously). When ϱ is a timed execution in a TA_ε, then the label is obtained by deleting from w the occurrences of pairs such that the first component is ε. If in addition ϱ is finite and ends in a final location, we say that the above timed word is accepted by $\mathcal{A}$. We denote $L(\mathcal{A})$ the set of finite timed words accepted by $\mathcal{A}$. Note that in a deterministic TA, every word has at most a single underlying timed execution.

Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ be a timed language. It is said *timed regular* whenever it is accepted by some TA, and ε-*timed regular* whenever it is accepted by some TA_ε. Note that if $\mathcal{L}$ is a(n ε-)timed regular language, then *Untimed*$(\mathcal{L})$ is also regular [4].

In this paper, we assume the reader is familiar with the region automaton construction and its properties, proposed by Alur and Dill in [3, 4].

2.3. Classical results on TA and TA_ε

We summarize all expressiveness and (un)decidability results we will use in our proofs. They only hold in the context of finite timed words. Similar (but slightly different) results will be presented in section 7.

Theorem 2.1. (Closure and expressiveness results)

1. The family of timed regular languages is not closed under complementation [4].

[2] The valuation $\vec{0}$ maps each clock to 0.

2. The family of timed regular languages accepted by deterministic TA is strictly included in the family of timed regular languages [4].

3. The family of timed regular languages is strictly included in the family of ε-timed regular languages [9].

The results of theorem 2.1 also hold for infinite words. This is not the case of the next theorem. The result for infinite words is stated by theorem 7.1 in section 7.

Theorem 2.2. (Universality problem)

1. The universality problem for TA is undecidable even when restricting to TA with two clocks [4] or with a one-letter alphabet [2].

2. The universality problem for TA with a single clock is decidable [22].

3. The universality problem for TA_ε with a single clock is undecidable [20].

2.4. Accepting timed words in TA and TA_ε

In this subsection, we give two examples of TA_ε which explain major difficulties that may arise from silent transitions.

Example 2.1. The TA_ε of Figure 1 recognizes the timed language

$$\mathcal{R}_{\mathit{even}} = \{(a,\tau_1)(a,\tau_2)\ldots(a,\tau_n) \mid \tau_i \equiv 0 \mod 2 \text{ for every } 1 \leq i \leq n\}\,.$$

This timed language is not recognized by any TA [9].

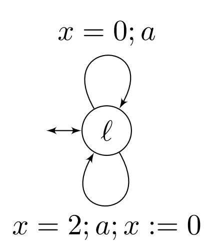

Figure 1. A TA_ε not equivalent to any TA

Example 2.2. The TA_ε of Figure 2 recognizes the timed language reduced to a singleton $\{(a,1)\}$. However any path $(\ell_0,0) \xrightarrow{d} (\ell_0,d) \xrightarrow{\varepsilon} (\ell_1,d) \xrightarrow{1-d} (\ell_1,1) \xrightarrow{a} (\ell_2,1)$, with $d \in [0,1]$, is an accepting path. Along one of these paths, denoted ϱ, configuration $(\ell^-_{\varrho,1}, v^-_{\varrho,1}) = (\ell_1,1)$ and $(\ell^+_{\varrho,1}, v^+_{\varrho,1}) = (\ell_2,0)$.

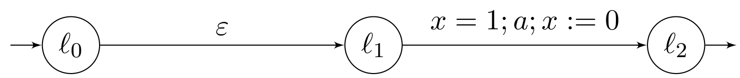

Figure 2. $(a, 1)$ is accepted by infinitely (and even uncountably) many paths

3. Removing Silent Transitions

In [9] the impact of silent transitions on the expressive power of timed automata has been studied, and syntactical restrictions have been given, that are sufficient to remove silent transitions, *i.e.*, syntactical restrictions for an ε-timed regular language to be timed regular. However, these syntactical restrictions are not necessary, and we prove in this section that the problem to decide whether an ε-timed regular language is timed regular is indeed undecidable.

Theorem 3.1. (Removing silent transitions)
Given a TA_ε $\mathcal{A}$, it is undecidable to determine whether there exists a TA $\mathcal{B}$ such that $L(\mathcal{A}) = L(\mathcal{B})$.

To prove this result, and other theorems in the sequel, we reduce the problem to the universality problem for timed automata. We first describe a construction over timed languages introduced by Finkel [18].

In the sequel, Σ denotes an alphabet, and c a fresh letter not in Σ. We set $\Sigma_+ = \Sigma \cup \{c\}$.

Definition 3.1. Let $\mathcal{L}$ and $\mathcal{R}$ be two timed languages over Σ. Then *Compose*$(\mathcal{L}, \mathcal{R})$ is a timed language over Σ_+ defined as the union of the following three languages:

$$\begin{aligned}
\mathcal{V}_1 &= \{w \in \mathcal{TW}^*(\Sigma_+) \mid \exists w' \in \mathcal{L},\ \exists w'' \in \mathcal{TW}^*(\Sigma), \exists \tau \text{ s.t. } w = w'(c, \tau)w''\} \\
\mathcal{V}_2 &= \{w \in \mathcal{TW}^*(\Sigma_+) \mid |w|_c \neq 1\} \\
\mathcal{V}_3 &= \{w \in \mathcal{TW}^*(\Sigma_+) \mid \exists w' \in \mathcal{TW}^*(\Sigma),\ \exists w'' \in \mathcal{R}, \exists \tau \text{ s.t. } w = w'(c, \tau)(w'' + \tau)\}
\end{aligned}$$

Now we state two fundamental properties of this construction that will be extensively used in the proofs.

Lemma 3.1. Let $\mathcal{L}$ and $\mathcal{R}$ be two timed languages over alphabet Σ.

- *Compose*$(\mathcal{TW}^*(\Sigma), \mathcal{R}) = \mathcal{TW}^*(\Sigma_+)$, it is thus accepted by a deterministic TA with no clock.
- If $\mathcal{L}$ and $\mathcal{R}$ are accepted by a TA_ε with at most n clocks, then *Compose*$(\mathcal{L}, \mathcal{R})$ is also accepted by a TA_ε with at most n clocks.

Proof:
The first point is a simple consequence of the definitions. We detail the proof of the second point. Let us denote $\mathcal{A}_\mathcal{L}$ (resp. $\mathcal{A}_\mathcal{R}$) a TA_ε accepting $\mathcal{L}$ (resp. $\mathcal{R}$) with at most n clocks. By definition, these automata have exactly one initial location and may have several final locations. We denote $\mathcal{A}$ the TA_ε obtained from $\mathcal{A}_\mathcal{L}$ and $\mathcal{L}_\mathcal{R}$ as depicted on Figure 3. Observe that in this construction, clocks of $\mathcal{A}_\mathcal{L}$ can be reused in $\mathcal{A}_\mathcal{R}$ since the two automata are not connected. Then $\mathcal{A}$ has at most n clocks, denoted by X, as requested. It is routine to verify that $\mathcal{A}$ accepts the language *Compose*$(\mathcal{L}, \mathcal{R})$. □

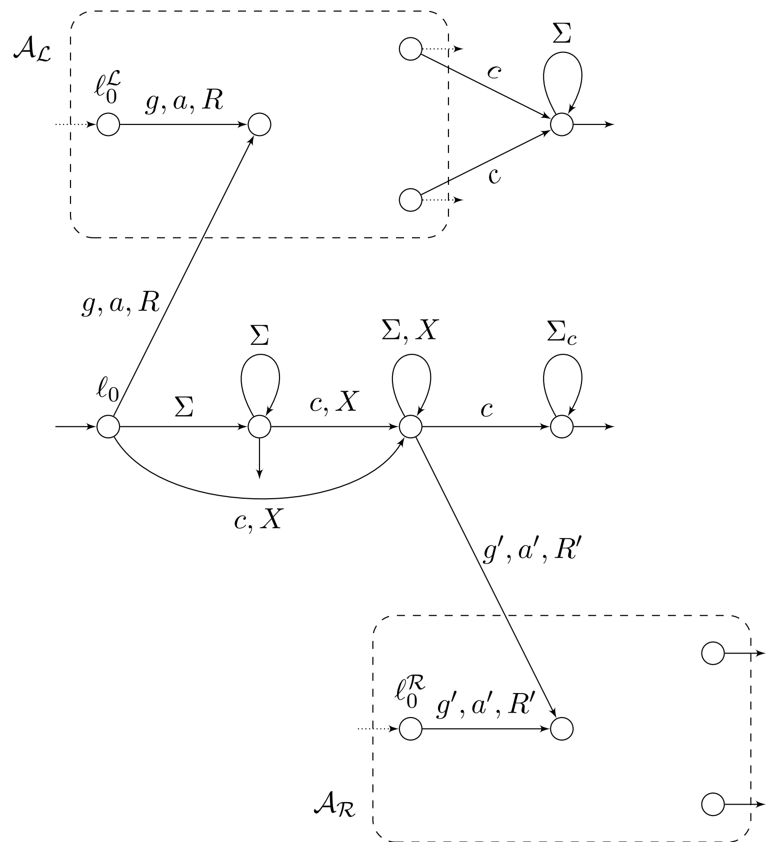

Figure 3. Automaton recognizing the language *Compose*$(\mathcal{L}, \mathcal{R})$.

Lemma 3.2. Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ be a timed regular language. Then *Compose*$(\mathcal{L}, \mathcal{R}_{even})$ is timed regular iff $\mathcal{L}$ is universal, where $\mathcal{R}_{even}$ is the timed language introduced in Subsection 2.4.

Proof:
We write $\mathcal{V} = Compose(\mathcal{L}, \mathcal{R}_{even})$. We will show now that $\mathcal{V}$ is timed regular if and only if $\mathcal{L}$ is universal on Σ. We distinguish two cases:

(1) **First case.** Assume $\mathcal{L} = \mathcal{TW}^*(\Sigma)$. Applying Lemma 3.1, $\mathcal{V} = \mathcal{TW}^*(\Sigma_+)$, which is obviously timed regular.

(2) **Second case.** Assume $\mathcal{L} \neq \mathcal{TW}^*(\Sigma)$. Towards a contradiction, assume that $\mathcal{V}$ is recognized by a TA $\mathcal{A}$. Let $y = (a_0, \tau_0) \ldots (a_n, \tau_n) \in \mathcal{TW}^*(\Sigma) \setminus \mathcal{L}$. Then we have that, for every $w \in \mathcal{TW}^*(\Sigma)$, $y.(c, \tau_n).(w + \tau_n) \in \mathcal{V}$ if and only if $w \in \mathcal{R}_{even}$. Mimicking the proof of [9] which shows that $\mathcal{R}_{even}$ is not timed regular, we will get a contradiction. Let K be the maximal constant of $\mathcal{A}$ and consider the timed word $w' = y.(c, \tau_n).(a, \tau + \tau_n)$ where $\tau \in \mathbb{N}$ is an even integer satisfying $\tau > K$. Then, the timed word w' is accepted by $\mathcal{A}$, and there exists a path in $\mathcal{A}$ along which w' is accepted. In particular, the last transition of this path, say $(\ell, \gamma, a, U, \ell')$, is such that $\ell' \in F$ is a final location. Let denote by (ℓ, v) the configuration reached after $y.(c, \tau_n)$ is recognized. Then

$v' = v + \tau$ is the valuation when firing the last transition, and verifies $v' \models \gamma$. Because of the choice of τ, it holds for any clock x of $\mathcal{A}$ that $v'(x) = v(x) + \tau > K$. In particular, for any odd integer τ' greater than τ, the timed word $y.(c, \tau_n).(a, \tau_n + \tau')$ is also accepted by $\mathcal{A}$, which is a contradiction. Hence, $\mathcal{V}$ cannot be recognized by a TA.

This concludes the proof: $\mathcal{L}$ is universal if and only if $\mathcal{V}$ is timed regular. □

We can now give the proof of Theorem 3.1.

Proof:
We assume that $a \in \Sigma$, and consider the timed language $\mathcal{R}_{even}$ introduced in Subsection 2.4. Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ be a timed regular language. The language $\mathcal{R}_{even}$ is ε-timed regular. Applying Lemma 3.1, we have that $\mathcal{V} = \mathit{Compose}(\mathcal{L}, \mathcal{R}_{even})$ is ε-timed regular. Applying lemma 3.2, we have that $\mathcal{V}$ is timed-regular iff $\mathcal{L}$ is universal. Thus the universality problem is reducible to the checking the timed regularity of an ε-timed regular language thus yielding the undecidability of the latter problem. □

4. Complementability and Determinizability

In [18, Theorem 1], Finkel proved that the problems whether the complement of a regular timed language is regular and whether a regular timed language can be recognized by a deterministic TA are both undecidable. We extend those results to the class TA_ε of timed automata with silent transitions.

Theorem 4.1. (Determinization)
It is undecidable to determine whether, for a given TA_ε $\mathcal{A}$, there exists a deterministic TA $\mathcal{B}$ such that $L(\mathcal{B}) = L(\mathcal{A})$.

Since TA are less expressive than TA_ε, the above result is a straightforward consequence of Finkel's result.

Theorem 4.2. (Complementation)

1. It is undecidable to determine whether, for a given TA_ε $\mathcal{A}$, there exists a TA_ε $\mathcal{B}$ such that $L(\mathcal{B}) = \overline{L(\mathcal{A})}$.
2. Furthermore this result holds for TA_ε over alphabets with two letters.

The proof of this theorem is neither a corollary of that of Finkel for the class TA, nor an obvious twist of his proof. Indeed, his proof heavily relies on the fact that given a timed word and a TA, there are *finitely* many timed executions wich yield such a timed word. This is no more the case for the class TA_ε, as mentioned in Subsection 2.4. We propose two undecidability proofs for that result, the simplest one which holds for timed automata over alphabets with three letters or more, and the other one, more involved, which holds for timed automata over alphabets with two letters.

The two proofs proceed as follows:

- Consider a regular timed language $\mathcal{L}$;
- Fix a regular timed language $\mathcal{R}$ such that $\overline{\mathcal{R}}$ is not regular;
- Build from $\mathcal{L}$ and $\mathcal{R}$, a new regular timed language $\mathit{Compose}(\mathcal{L}, \mathcal{R})$ (which has been defined in the previous section) such that $\mathcal{L}$ is universal iff the complement of $\mathit{Compose}(\mathcal{L}, \mathcal{R})$ is regular.

4.1. Case of TA_ε over alphabets with three letters or more

For this proof, we instantiate the language $\mathcal{R}$ by a language proposed in [6] for gracefully proving that the class of timed regular languages is not closed under complement. It turns out that their result, proved in the framework of timed regular languages, also holds in the framework of ε-timed regular languages, as stated in the following proposition.

Proposition 4.1. Assume $\Sigma = \{a, b\}$, and let $\mathcal{R}_{a,b}$ be the timed language

$$\mathcal{R}_{a,b} = \{w = (a_0, \tau_0) \dots (a_n, \tau_n) \in \mathcal{TW}^*(\Sigma) \mid \exists i,\ a_i = a, \text{ and } \forall j \geq i,\ \tau_j - \tau_i \neq 1\}.$$

This timed language is timed regular, but its complement is not ε-timed regular.

The proof of this proposition is similar to that in [6], but for sake of completeness, we write it there as well.

Proof:
The timed language $\mathcal{R}_{a,b}$ is accepted by the timed automaton depicted on Figure 4, hence it is timed regular. We now show that its complement is not ε-timed regular, *i.e.*, that it cannot be recognized by

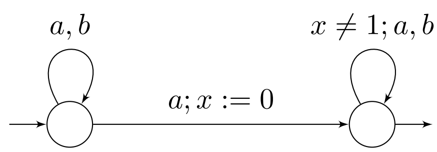

Figure 4. The timed automaton accepting $\mathcal{R}_{a,b}$

any TA_ε. Assume that there exists a TA_ε $\mathcal{B}$ such that $L(\mathcal{B}) = \overline{\mathcal{R}_{a,b}}$. The complement of $\mathcal{R}_{a,b}$ is the set of timed words in which every action a is followed one time unit later by an action.

Let $\mathcal{T}_1$ be the set of timed words w over Σ such that:

(i) *Untimed*(w) belongs to the untimed regular language a^*b^*,

(ii) all a's occur within $[0, 1[$, and

(iii) no two a's occur at the same date.

It is straightforward to check that $\mathcal{T}_1$ is timed regular. Now observe that a word of the form $a^n b^m$ belongs to *Untimed*$(\mathcal{T}_1 \cap \overline{\mathcal{R}_{a,b}})$ if and only if $m \geq n$ holds. Hence a contradiction: both intersection and the *Untimed* operator preserve regularity of languages, and $\{a^n b^m \mid m \geq n\}$ is not regular. □

The following lemma will be useful on the proof of Theorem 4.2.1. This is the counterpart of [4, Theorem 3.17] for complements of timed regular languages.

Lemma 4.1. Let $\mathcal{A}$ be a TA_ε over alphabet Σ and $w \notin L(\mathcal{A})$ be a finite timed word, then there is another timed word $w' \notin L(\mathcal{A})$ whose dates are rational. Furthermore, *Untimed*$(w) =$ *Untimed*(w').

Proof:
Let d be the granularity of $\mathcal{A}$. Let $w = (a_1, \tau_1) \ldots (a_n, \tau_n)$. For convenience of notations, we define $\tau_0 = 0$. We build $w' = (a_1, \tau_1') \ldots (a_n, \tau_n')$ by induction. Moreover the timed word will satisfy this property:

$$\forall 0 \leq i < j \leq n, \forall k \in \mathbb{N}, \tau_j - \tau_i \sim k/d \Leftrightarrow \tau_j' - \tau_i' \sim k/d \text{ with } \sim \in \{<, \leq\} \tag{1}$$

The inductive property is the following one: there is a word $w^m = (a_1, \tau_1^m) \ldots (a_n, \tau_n^m)$ fulfilling the property (1) with $\forall i \leq m, \tau_i^m \in \mathbb{Q}$. The base case is proved by taking $w^0 = w$.

Assume that there is a word $w^m = (a_1, \tau_1^m) \ldots (a_n, \tau_n^m)$ fulfilling property (1). If $\tau_{m+1}^m \in \mathbb{Q}$ then $w^{m+1} = w^m$. Otherwise we split the set of indexes $I = \{0, 1, \ldots, n\}$ in two subsets $I_\equiv = \{i \in I \mid \tau_i^m \equiv \tau_{m+1}^m \mod 1/d\}$ and $I_{\not\equiv} = I \setminus I_\equiv$. As $\tau_i^m \in \mathbb{Q}$ for all $i \leq m$ (by induction hypothesis) and $\tau_{m+1}^m \notin \mathbb{Q}$, we have that $\{0, \ldots, m\} \subseteq I_{\not\equiv}$. For each $i \in I_{\not\equiv}$, define δ_i as the distance between τ_i^m and the $1/d$-grid around τ_{m+1}^m: $\delta_i = \min(|\tau_i^m - \tau_{m+1}^m - k/d| \mid k \in \mathbb{Z})$. We then set δ as the minimum over $I_{\not\equiv}$ of these distances: $\delta = \min(\delta_i \mid i \in I_{\not\equiv})$. Observe that $\delta > 0$. Pick some δ' such that $0 < \delta' < \delta$ and $\tau_{m+1}^m + \delta' \in \mathbb{Q}$. We build w^{m+1} as follows. $\forall i \in I_{\not\equiv}, \tau_i^{m+1} = \tau_i^m$ and $\forall i \in I_\equiv, \tau_i^{m+1} = \tau_i^m + \delta'$. It is easy to check that w^{m+1} fulfills the inductive property. Indeed, we first have that $\tau_i^{m+1} \in \mathbb{Q}$ for $i \leq m+1$ by the choice of δ' and by the inductive property. Second, if i and j both belong to $I_\equiv$ or $I_{\not\equiv}$, then the value of $\tau_j - \tau_i$ is unchanged, and otherwise it is incremented (or decremented) of δ', what does not change the validity of property (1) by the choice of δ'.

We claim that $w' \notin L(\mathcal{A})$. By contradiction, assume that $w' \in L(\mathcal{A})$ and let $(\ell_0, v_0) \xrightarrow{\tau_1'} (\ell_0, v_0 + \tau_1') \xrightarrow{e_1} (\ell_1, v_1) \ldots (\ell_{n-1}, v_{n-1} + \tau_n' - \tau_{n-1}') \xrightarrow{e_n} (\ell_n, v_n)$ be a finite accepting path for w'. Examine the path $(\ell_0, v_0) \xrightarrow{\tau_1} (\ell_0, v_0 + \tau_1) \xrightarrow{e_1} (\ell_1, v_1) \ldots (\ell_{n-1}, v_{n-1} + \tau_n - \tau_{n-1}) \xrightarrow{e_n} (\ell_n, v_n)$. The value of a clock x when firing an edge e_i in the former path is $\tau_i' - \tau_j'$ for some $j < i$ (corresponding to the last reset of x before firing e_i) and this value in the latter path is $\tau_i - \tau_j$. Due to property (1) on time differences relative to w and w', the previous observation shows that the guard of every e_i in the latter path is satisfied and thus $w \in L(\mathcal{A})$ which yields a contradiction. $\square$

We can now prove the following Lemma, from which Theorem 4.2.1 easily follows since universality of timed automata is undecidable.

Lemma 4.2. Assume $\{a, b\} \subseteq \Sigma$. Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ be a timed regular language, and define the timed language $\mathcal{V}$ over Σ_+ as $\mathcal{V} = \mathit{Compose}(\mathcal{L}, \mathcal{R}_{a,b})$. Then $\overline{\mathcal{V}}$ is recognized by a TA_ε if, and only if, $\mathcal{L}$ is universal.

Proof:
To prove this lemma, we distinguish two cases:

(1) **First case.** Assume $\mathcal{L} = \mathcal{TW}^*(\Sigma)$. Applying Lemma 3.1, $\mathcal{V} = \mathcal{TW}^*(\Sigma_+)$. Thus, $\overline{\mathcal{V}} = \emptyset$, which is obviously ($\varepsilon$-)timed regular.

(2) **Second case.** Assume $\mathcal{L} \neq \mathcal{TW}^*(\Sigma)$. Towards a contradiction, assume that $\overline{\mathcal{V}}$ is recognized by a TA_ε $\mathcal{A}'$ with granularity d. Let $w = (a_0, \tau_0) \ldots (a_p, \tau_p) \in \mathcal{TW}^*(\Sigma) \setminus \mathcal{L}$. By Lemma 4.1, we can

assume that all dates are rational. We define the timed regular language $\mathcal{T}_2$ as follows:

$$w' = w(c,\tau_p)x \in \mathcal{T}_2 \iff \begin{cases} \mathit{Untimed}(x) \in a^*b^*, \\ x = (a,\tau'_0)\dots(a,\tau'_{k_1})(b,\tau''_0)\dots(b,\tau''_{k_2}), \\ \forall 0 \le i \le k_1,\ \tau'_i \in [\tau_p, \tau_p+1[, \\ \forall 0 \le i \neq j \le k_1,\ \tau'_i \neq \tau'_j\,. \end{cases}$$

Similarly to the proof of Proposition 4.1, observe that $\mathit{Untimed}(\mathcal{T}_2 \cap \overline{\mathcal{V}}) = \{w' \mid \exists m \ge n, w' = \mathit{Untimed}(w)\, c\, a^n b^m\}$. This contradicts our assumption that $\overline{\mathcal{V}}$ be ε-timed regular since the right member of the previous equality is not regular.

This concludes the proof: $\mathcal{L}$ is universal iff $\overline{\mathcal{V}}$ is ε-timed regular. □

4.2. Case of TA_ε over alphabets with two letters

We first state the following lemma:

Lemma 4.3. Let $\mathcal{A}$ be a TA_ε with n clocks and granularity d. Let (ℓ, v) be a configuration of $\mathcal{A}$. Then $[0, 1/d[$ can be partitioned as $I_1 \cup \ldots \cup I_m$ where $I_1, \ldots, I_m$ are disjoint consecutive intervals such that $m \le 2n+1$, and for every $1 \le j \le m$, for all $\delta, \delta' \in I_j$, for every $k \in \mathbb{N}$, for all $x \in X$, for every $\bowtie\ \in \{<, \le\}$,

$$v(x) + \delta \bowtie k/d \iff v(x) + \delta' \bowtie k/d\,.$$

Proof:
For every $x \in X$, there is exactly one value $\delta_x \in [0, 1/d[$ such that $v(x)+\delta_x \in \mathbb{N}_d$. Let $\Delta = \{\delta_1, \ldots, \delta_J\}$ be the set of such values, assuming $\delta_i < \delta_{i+1}$. Of course, $J \le n$. Let the partition $[0, 1/d[$ be given by $[0, \delta_1[\uplus [\delta_1, \delta_1] \uplus]\delta_1, \delta_2[\uplus \ldots \uplus]\delta_J, 1/d[$. It is routine to check that this partition fulfills the requirement of the lemma. □

The next proposition extends to TA_ε the well-known result [4] that the class of TA over an alphabet reduced to a singleton is not closed under complementation.

Proposition 4.2. Let $\mathcal{R}_a$ be the following timed language:

$$\mathcal{R}_a = \{(a,\tau_1)\ldots(a,\tau_n) \mid \exists 1 \le i < j \le n \text{ s.t. } \tau_j - \tau_i = 1\}\,.$$

Then $\overline{\mathcal{R}_a}$ is not ε-timed regular.

Proof:
Towards a contradiction, we assume that the TA_ε $\mathcal{A}$ recognizes the language $\overline{\mathcal{R}_a}$. We denote by n the number of clocks of $\mathcal{A}$, and by d its granularity. The language $\overline{\mathcal{R}_a}$ is the set of timed words such that no pair of occurrences of a's are separated by one time unit.

Pick a timed word $w = (a,\tau_1)\ldots(a,\tau_{2N+1})$ in $\overline{\mathcal{R}_a}$ such that $N = 2n+1$, and:

- for all $1 \leq i < j \leq N$, $0 < \tau_i < \tau_j < 1/d$,
- for all $1 \leq i \leq N$, $1 < \tau_{N+i} < 1 + \tau_i < \tau_{N+i+1} < 1 + 1/d$.

Let ϱ be a timed execution of $\mathcal{A}$ which accepts w, and consider the configuration $(\ell^-_{\varrho,1}, v^-_{\varrho,1})$. Applying Lemma 4.3 to this configuration, we get a partition of $[0, 1/d[$ composed of at most N intervals. There exists a j such that $N + 1 \leq j \leq 2N$, $\tau_j - 1$ and $\tau_{j+1} - 1$ belong to the same interval of this partition.

We now prove that for each $x \in X$, there exists $k \in \mathbb{N}$ such that:

$$k/d < v^-_{\varrho,\tau_j}(x) < v^-_{\varrho,\tau_j}(x) + (\tau_{j+1} - \tau_j) < (k+1)/d \tag{2}$$

Let $x \in X$. We distinguish two cases:

- Assume that x has not been reset between the configurations $(\ell^-_{\varrho,1}, v^-_{\varrho,1})$ and $(\ell^-_{\varrho,\tau_j}, v^-_{\varrho,\tau_j})$ along ϱ. It implies that $v^-_{\varrho,\tau_j}(x) = v^-_{\varrho,1}(x) + \tau_j - 1$. Due to the choice of j, we know that $v^-_{\varrho,1}(x) + \tau_j - 1$ and $v^-_{\varrho,\tau_j}(x) + (\tau_{j+1} - \tau_j) = v^-_{\varrho,1}(x) + \tau_{j+1} - 1$ satisfy the same constraints of 'granularity d'. Hence, equation (2) holds for clock x.
- Assume that x has been reset along ϱ between the configurations $(\ell^-_{\varrho,1}, v^-_{\varrho,1})$ and $(\ell^-_{\varrho,\tau_j}, v^-_{\varrho,\tau_j})$. In this case, equation (2) holds for $k = 0$. Indeed $0 < v^-_{\varrho,\tau_j}(x)$ since $\tau_j > 0$. Furthermore, as the date at which clock x has been last reset between the two above-mentioned configurations is within the interval $[1, 1 + 1/d[$, we get that $v^-_{\varrho,\tau_j}(x) + (\tau_{j+1} - \tau_j) \leq (\tau_j - 1) + (\tau_{j+1} - \tau_j) < 1/d$.

Let $\delta = 1 + \tau_{j-N} - \tau_j$. From equation (2) and the constraints on the time sequence $(\tau_i)_{1 \leq i \leq 2N+1}$, we get that for every $x \in X$, there exists some $k \in \mathbb{N}$ such that:

$$k/d < v^-_{\varrho,\tau_j}(x) < v^-_{\varrho,\tau_j}(x) + \delta < (k+1)/d \tag{3}$$

Now we build a timed execution ϱ' as follows. It mimics ϱ up to the configuration $(\ell^-_{\varrho,\tau_j}, v^-_{\varrho,\tau_j})$. Then, it lets δ time units elapse, which leads to configuration $(\ell^-_{\varrho,\tau_j}, v^-_{\varrho,\tau_j} + \delta)$. Then it fires the *instantaneous* subsequence (*i.e.*, with null duration) of ϱ (say ϱ_j) leading from $(\ell^-_{\varrho,\tau_j}, v^-_{\varrho,\tau_j})$ to $(\ell^+_{\varrho,\tau_j}, v^+_{\varrho,\tau_j})$. The timed execution ϱ_j is non empty as it contains at least a transition labelled by a. This sequence can also be executed from $(\ell^-_{\varrho,\tau_j}, v^-_{\varrho,\tau_j} + \delta)$ since, following equation (3), both configurations satisfy the same constraints of granularity d. More precisely, using the notion of 'regions' associated with $\mathcal{A}$ (we refer to [4] for a definition and properties of regions in timed automata), the two configurations belong to the same region, and so do the two configurations reached after firing ϱ_j. Then, due to the so-called time-abstract bisimulation property of regions, it is possible to extend ϱ' from this configuration by the same actions as ϱ, possibly with other delays, until reaching an accepting location (as ϱ is accepting).

Now, the timed word read on ϱ' has two occurrences of a separated by one time unit (those at date τ_{j-N} and at date $1 + \tau_{j-N}$). Thus, it does not belong to $\overline{\mathcal{R}_a}$, hence a contradiction. $\square$

We can now prove the following Lemma from which easily follows Theorem 4.2.2, since universality of timed automata over a one-letter alphabet is undecidable (recall Theorem 2.2, item 1).

Lemma 4.4. Assume $a \in \Sigma$. Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ be a timed regular language, and define the timed language $\mathcal{V}$ over Σ_+ as $\mathcal{V} = \textit{Compose}(\mathcal{L}, \mathcal{R}_a)$. Then $\overline{\mathcal{V}}$ is recognized by a TA_ε if, and only if, $\mathcal{L}$ is universal.

Proof:
We distinguish two cases:

(1) **First case.** We assume $\mathcal{L} = \mathcal{TW}^*(\Sigma)$. As a consequence of Lemma 3.1, $\overline{\mathcal{V}} = \emptyset$ which is obviously (ε-)timed regular.

(2) **Second case.** We assume $\mathcal{L} \neq \mathcal{TW}^*(\Sigma)$. Towards a contradiction, we assume that $\overline{\mathcal{V}}$ is recognized by a TA_ε $\mathcal{A}'$ with granularity d and n clocks. Pick $w' = (a_1, \tau'_1) \dots (a_m, \tau'_m)$ in $\mathcal{TW}^*(\Sigma) \setminus \mathcal{L}$ and let $w = w'(c, \tau'_m)(a, \tau_1) \dots (a, \tau_{2N+1}) \in \mathcal{V}$ with $N = 2n + 1$ such that:

- for all $1 \leq i < j \leq N$, $\tau'_m < \tau_i < \tau_j < \tau'_m + 1/d$;
- for all $1 \leq i \leq N$, $\tau'_m + 1 < \tau_{N+i} < 1 + \tau_i < \tau_{N+i+1} < \tau'_m + 1 + 1/d$.

From a timed execution accepting w in $\mathcal{A}$, we construct a timed execution ϱ' (which plays the words w', the N next actions, and then applies the construction of the proof of Proposition 4.2 from $(\ell^-_{\varrho,\tau'_m+1}, v^-_{\varrho,\tau'_m+1})$) to obtain another accepting execution ϱ' whose associated word does not belong to $\mathcal{V}$, yielding a contradiction.

This concludes the proof: $\mathcal{L}$ is universal iff $\overline{\mathcal{V}}$ is ε-timed regular. □

5. Minimization of the Number of Clocks

In [18, Theorem 2], Finkel proved that given a timed language recognized by a TA with n clocks ($n \geq 2$), we cannot decide whether it can be recognized by a TA with $n-1$ clocks. In this Section, we prove that this result also holds in the framework of TA_ε.

We first prove the following proposition, which exhibits a family of timed languages such that the n-th language is recognized by an n-clock TA, but not by any $(n-1)$-clock TA_ε. These languages are known since [19] when restricting to TA. However, the extension of the result to TA_ε is non-trivial since silent transitions allow more complex timed executions (see subsection 2.4).

Proposition 5.1. (Language with a minimal number of clocks)
Let $n \geq 1$ be a positive integer. Define the language $\mathcal{R}_n$ as follows:

$$\mathcal{R}_n = \{(a, \tau_1)(a, \tau_2) \dots (a, \tau_{2n}) \mid \forall 1 \leq i \leq n,\ 0 \leq \tau_i < 1 \wedge \tau_{n+i} = 1 + \tau_i\}.$$

The timed language $\mathcal{R}_n$ is accepted by a TA with n clocks, but not by any TA, and even any TA_ε, with strictly less than n clocks.

Proof:
Let $n \geq 1$ be a positive integer. The language $\mathcal{R}_n$ is recognized by the TA $\mathcal{A}_n$ depicted on Figure 5.

Now assume that there exists a TA_ε $\mathcal{B}$ with less than n clocks, and such that $L(\mathcal{B}) = L(\mathcal{A})$. Denote by d the granularity of $\mathcal{B}$. Fix some values $(\tau_i)_{1 \leq i \leq n}$ such that $0 < \tau_1 < \tau_2 < \dots < \tau_n < 1/d$, and consider the timed word $w = (a, \tau_1)(a, \tau_2) \dots (a, \tau_n)(a, \tau_1 + 1)(a, \tau_2 + 1) \dots (a, \tau_n + 1)$. Obviously $w \in \mathcal{R}_n$, and thus w is accepted by $\mathcal{B}$ along some run ϱ.

For each index i in $\{1, \dots, n\}$, we consider the configuration $(\ell^-_{\varrho,\tau_i+1}, v^-_{\varrho,\tau_i+1})$. The last transition before this configuration is thus a delay transition. We distinguish two cases:

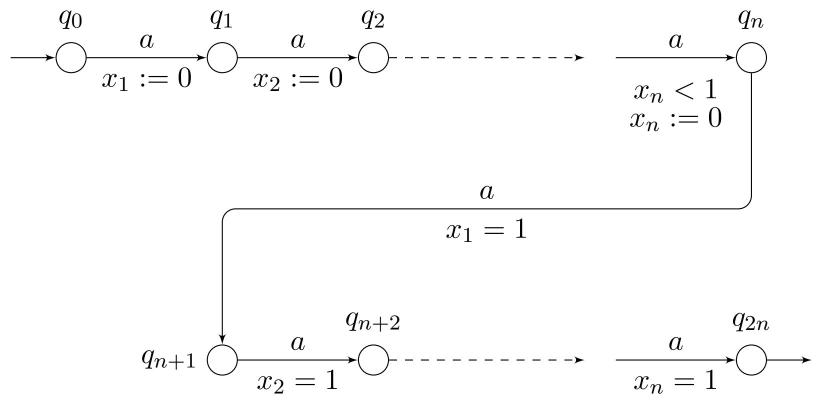

Figure 5. Automaton $\mathcal{A}_n$ with n clocks.

- **First case:** There exists an index $i \in \{1, \ldots, n\}$ such that for every clock x, $v^-_{\varrho,\tau_i+1}(x) \not\equiv 0 \mod 1/d$. This implies that the region (with respect to granularity d) to which belongs the valuation v^-_{ϱ,τ_i+1} is 'time-open', *i.e.*, for every $v \in r$, there exists $\delta > 0$ such that $v + \delta \in r$ and $v - \delta \in r$. Thus, we can change the time elapsed during the last transition, and add such a value δ. The new configuration which is reached along this modified execution is $(\ell^-_{\varrho,\tau_i+1}, v^-_{\varrho,\tau_i+1} + \delta)$ and $v^-_{\varrho,\tau_i+1} + \delta$ is in the same region as v^-_{ϱ,τ_i+1}. Hence, applying the time-abstract bisimulation property of the regions, it is possible to follow exactly the same transitions (possibly at different dates). This gives another accepting execution. Nevertheless, the timed word which is read on this execution does not belong to $\mathcal{R}_n$ because the i-th a and the $i + N$-th a are separated by $1 + \delta > 1$ units of time. Hence a contradiction.

- **Second case:** We assume that for every index $i \in \{1, \ldots, n\}$, there exists a clock x such that $v^-_{\varrho,\tau_i+1}(x) \equiv 0 \mod 1/d$. Since the number of clocks of $\mathcal{B}$ is strictly less than n, there exists a clock x such that $v^-_{\varrho,\tau_i+1}(x) \equiv 0 \mod 1/d$ and $v^-_{\varrho,\tau_j+1}(x) \equiv 0 \mod 1/d$ with $1 \leq i < j \leq n$. Since $\tau_i + 1 > 0$ and $\tau_j + 1 > 0$, the two values $v^-_{\varrho,\tau_i+1}(x)$ and $v^-_{\varrho,\tau_j+1}(x)$ are positive, hence some k/d for $k \in \mathbb{N}^*$. This leads to a contradiction, as the time elapsed between these two positions is strictly less than $1/d$ (and positive).

This concludes the proof: such a TA_ε $\mathcal{B}$ cannot exist. □

We first establish the following property of the construction *Compose* applied to languages R_n.

Lemma 5.1. Assume $a \in \Sigma$ and n be an integer.

- **Case** $n \geq 2$**.** Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ accepted by some TA with n clocks.

- **Case** $n \geq 1$**.** Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ accepted by some TA_ε with 1 clock.

Consider $\mathcal{V}_n$ over $\Sigma_+ = \Sigma \cup \{c\}$ defined as $\mathcal{V}_n = \mathit{Compose}(\mathcal{L}, \mathcal{R}_n)$. Then $\mathcal{L}$ is universal if, and only if, $\mathcal{V}_n$ is accepted by a TA_ε with $n - 1$ clocks.

Proof:
We distinguish two cases:

1. **First case.** We assume $\mathcal{L}$ is universal on Σ, *i.e.* $\mathcal{L} = \mathcal{TW}^*(\Sigma^*)$. Then, $\mathcal{V}_n = \mathcal{TW}^*(\Sigma_+)$, *i.e.*, $\mathcal{V}_n$ is universal on Σ^+, and thus it can be accepted by a (deterministic) timed automaton *without any clock*.

2. **Second case.** We assume $\mathcal{L}$ is not universal on Σ, *i.e.*, $\mathcal{L}$ is strictly included in $\mathcal{TW}^*(\Sigma)$. Then, there is a timed word $u = (a_1, \tau_1) \ldots (a_k, \tau_k) \in \mathcal{TW}^*(\Sigma)$ which does not belong to $\mathcal{L}$. Consider now a timed word $x \in \mathcal{TW}^*(\Sigma)$. It holds that $u.(c, \tau_k).(x + \tau_k) \in \mathcal{V}_n$ iff $x \in \mathcal{R}_n$. Towards a contradiction, assume that $\mathcal{V}_n$ is accepted by a TA_ε $\mathcal{B}$ with $n-1$ clocks. Let us denote by d the granularity of $\mathcal{B}$, and fix some values $(\tau'_i)_{1 \le i \le n}$ such that $0 < \tau'_1 < \tau'_2 < \ldots < \tau'_n < 1/d$. We consider the timed word $v = (a, \tau'_1)(a, \tau'_2) \ldots (a, \tau'_n)(a, \tau'_1 + 1)(a, \tau'_2 + 1) \ldots (a, \tau'_n + 1)$. Obviously, $v \in \mathcal{R}_n$, and thus $w = u.(c, \tau_k).(v + \tau_k) \in \mathcal{V}_n$ is accepted by $\mathcal{B}$. We can then apply the reasoning developed in the proof of Proposition 5.1 to the timed word w, and get a contradiction. Indeed, this proof does not rely on the fact that the initial valuation is $\vec{0}$ and thus can be reproduced from configuration reached after recognizing $u.(c, \tau_k)$. We can finally conclude that such a timed automaton $\mathcal{B}$ cannot exist. Hence, the timed language $\mathcal{V}_n$ cannot be recognized by any TA_ε with strictly less than n clocks.

Thus determining whether $\mathcal{V}_n$ can be recognized by a TA_ε with less than n clocks is equivalent to deciding whether $\mathcal{L}$ is universal. □

We can now state the following theorem, which extends Theorem 2 of [18] to timed automata with silent transitions. Note that our undecidability result holds even for one-clock TA_ε, in contrast with the class of one-clock TA for which we can decide this problem.

Theorem 5.1. (Minimizing the number of clocks)
Let n be an integer.

- **Case $n \ge 2$.** For $n \ge 2$, it is undecidable to determine whether, for a given TA (and thus also for TA_ε) $\mathcal{A}$ with n clocks, there exists a TA_ε $\mathcal{B}$ with $n-1$ clocks such that $L(\mathcal{A}) = L(\mathcal{B})$.

- **Case $n \ge 1$.** It is undecidable to determine whether, for a given TA_ε $\mathcal{A}$ with 1 clock, there exists a TA_ε $\mathcal{B}$ without clocks such that $L(\mathcal{A}) = L(\mathcal{B})$.

Proof:
The proof follows from Lemma 5.1. Assume a timed language $\mathcal{L}$ given as described in Lemma 5.1. Due to Lemma 3.1, the timed language $\mathcal{V}_n$ is timed regular (resp. ε-timed regular when $n = 1$) and is accepted by a TA with n clocks (resp. a TA_ε with a single clock). Since the two universality problems that we consider are undecidable (see Theorem 2.2), this concludes the proof. □

6. Shuffle Operation

In this section, we are interested in the shuffle operation for timed words. In order to conform to the definition considered in [18] and in [16], we introduce a new description of timed words: given a timed word $w = (a_0, \tau_0) \ldots (a_n, \tau_n) \ldots$, we define its associated *delay timed word*, denoted $\mathsf{Delay}(w)$, and defined by

$$\mathsf{Delay}(w) = (\tau_0, a_0) \cdot ((\tau_1 - \tau_0), a_1) \cdots ((\tau_n - \tau_{n-1}), a_n) \cdots$$

Delay timed words are thus simply words on the alphabet $(\mathbb{R}_{\geq 0} \times \Sigma)$, *i.e.*, elements of $(\mathbb{R}_{\geq 0} \times \Sigma)^*$. This description of a timed word gathers the delay of time that elapses together with the next discrete action. Delay is a bijection between timed words ($\mathcal{TW}^*(\Sigma)$) and delay timed words ($(\mathbb{R}_{\geq 0} \times \Sigma)^*$).

We first define the shuffle operation on finite words on an alphabet X. Given $u, v \in X^*$, we define $u \sqcup\!\sqcup v$ as the set of words

$$\{w = x_1 y_1 x_2 y_2 \ldots x_n y_n \mid u = x_1 x_2 \ldots x_n \text{ and } v = y_1 y_2 \ldots y_n\}\,.$$

We extend it to sets of words by defining, for $S_1, S_2 \subseteq X^*$, $S_1 \sqcup\!\sqcup S_2 = \{s_1 \sqcup\!\sqcup s_2 \mid s_1 \in S_1, s_2 \in S_2\}$.

This definition thus directly applies to delay timed words (alphabet $X = (\mathbb{R}_{\geq 0} \times \Sigma)$) and *via* the Delay mapping can be used to define the shuffle operation $\sqcup\!\sqcup$ on timed words. Given u and v in $\mathcal{TW}^*(\Sigma)$,

$$u \sqcup\!\sqcup v = \mathsf{Delay}^{-1}(\mathsf{Delay}(u) \sqcup\!\sqcup \mathsf{Delay}(v))\,.$$

It also extends to delay timed languages, *i.e.*, sets of delay timed words, by previous definition on sets of words. Moreover, we define naturally the notions of (ε-) delay timed regular languages, as those associated with (ε-) timed regular languages by the operator Delay.

Remark 6.1. The shuffle operation corresponds to two tasks to be executed on the same processor by time sharing which have been extensively studied for monoprocessor scheduling problems.

In order to simplify the notations, in the sequel of this section, we only handle delay timed words. The results for timed words are obtained *via* the Delay mapping.

Finkel and Dima proved independently that delay timed regular languages are not closed under shuffle operation. We first extend this result, stated as [18, Theorem 4], to ε-delay timed regular languages.

Proposition 6.1. The shuffle of two delay timed regular languages is not necessarily an ε-delay timed regular language.

Proof:
To prove this result, we follow the lines of the proof of [18]. We first define three delay timed regular languages:

- $\mathcal{N}_1 = \{(t_1, a) \cdot (1, a) \cdot (t_2, a) \mid t_1 + t_2 = 1\}$,
- $\mathcal{N}_2 = \{(1, b) \cdot (s, b) \mid s \in \mathbb{R}_{\geq 0}\}$,
- $\mathcal{N}_3 = \{(t_1, a) \cdot (1, b) \cdot (s, b) \cdot (1, a) \cdot (t_2, a) \mid t_1, s, t_2 \in \mathbb{R}_{\geq 0}\}$.

If the shuffle of two delay timed regular languages was an ε-delay timed regular language, and since ε-delay timed regular languages are closed under intersection, the delay timed language $(\mathcal{N}_1 \sqcup\!\sqcup \mathcal{N}_2) \cap \mathcal{N}_3$ would also be ε-delay timed regular. We show that this is not the case.

$$(\mathcal{N}_1 \sqcup\!\sqcup \mathcal{N}_2) \cap \mathcal{N}_3 = \{(t_1, a) \cdot (1, b) \cdot (s, b) \cdot (1, a) \cdot (t_2, a) \mid t_1, t_2, s \in \mathbb{R}_{\geq 0},\ t_1 + t_2 = 1\}$$

Towards a contradiction, we suppose that there exists a TA_ε $\mathcal{A}$ accepting this language. We denote by d the granularity of $\mathcal{A}$.

Let w be a delay timed word accepted by $\mathcal{A}$ such that the following properties hold:

$$\begin{cases} t_2 & \not\equiv 0 \mod 1/d \\ s + t_2 & \not\equiv 0 \mod 1/d \\ s & \not\equiv 0 \mod 1/d \end{cases}$$

Since w is accepted by $\mathcal{A}$, there exists a path in the automaton which recognizes w, $\ell_0 \xrightarrow{e_1} \ell_1 \ldots \ell_{n-1} \xrightarrow{e_n} \ell_n$ where e_i are edges of $\mathcal{A}$. This path can be viewed as a (linear [3]) TA_ε $\mathcal{A}'$ with $n+1$ locations corresponding to the occurrences of locations in the path and n edges corresponding to the occurrences of edges in the path. The clocks of the two automata are the same ones. The guard and the reset of an occurring edge are the ones of the original one. The set of final locations is a singleton whose element corresponds to ℓ_n.

By construction, $\mathcal{A}'$ has no cycle, $w \in L(\mathcal{A}') \subseteq L(\mathcal{A})$ and its granularity d' divides the one of $\mathcal{A}$.

Using [9, Theorem 21], it is possible to build from $\mathcal{A}'$ another timed automaton without silent transitions $\mathcal{A}''$ accepting the same timed language, and such that its granularity is equal to that of $\mathcal{A}'$. Let us examine in the region automaton of $\mathcal{A}''$, a path which accepts w. Due to assumptions made on s, t_1 and t_2, the region reached immediately before the firing of the third a is time-open. Indeed a region is time-open as soon as there exists a clock valuation inside it such that every clock value is not equivalent to 0 modulo the granularity of the automaton. An elementary examination of the timed word yields to the possible clock values: $t_2, t_2+1, t_2+1+s, t_2+2+s, 3+s$ (recall that there are no silent transitions in $\mathcal{A}''$).

As a consequence, we can postpone the date at which this transition is taken by a small delay. We obtain another timed word w' which is accepted by $\mathcal{A}''$, but which does not satisfy the constraints of $(\mathcal{N}_1 \sqcup\!\sqcup \mathcal{N}_2) \cap \mathcal{N}_3$ (*i.e.*, $t_1 + t_2 = 1$). This yields a contradiction since $w' \in \mathcal{L}(\mathcal{A}'') = \mathcal{L}(\mathcal{A}') \subseteq \mathcal{L}(\mathcal{A})$. □

Observation. Let us analyze the scheme of the previous proof:

1. fix a word w in the language $\mathcal{L}$ under study;

2. transform one of its accepting paths into a linear TA_ε which accepts a language $\mathcal{L}'$ such that $w \in \mathcal{L}'$ and $\mathcal{L}' \subseteq \mathcal{L}$;

3. transform this linear TA_ε into a TA using the construction of [9] which accepts the same language $\mathcal{L}'$ (this is possible as there is no cycle in the TA_ε);

4. apply a technique specific to TA in order to obtain a word w' accepted by this TA such that $w' \notin \mathcal{L}$.

One could believe that such a scheme could be adapted to prove the previous results of this paper. However, it is worth noticing that the intricate construction of [9], when it is applied to silent transitions which reset some clocks, increases the number of clocks. This prevents the application of this scheme to the proofs which rely on the number of clocks of the original TA_ε (more precisely theorems 4.2.2 and 5.1).

[3] Linear TA_ε stands for a TA_ε with no cycle.

We now state our extension of [18, Theorem 5][4] to the framework of TA_ε.

Theorem 6.1. (Shuffle)
The problem of deciding whether the shuffle of two delay timed regular languages is ε-delay timed regular is undecidable.

Proof:
Let Σ be a finite alphabet containing at least one letter a. We denote by b and c two letters not in Σ, and define $\Sigma_+ = \Sigma \cup \{c\}$ and $\Sigma_b = \Sigma \cup \{b\}$. We consider a delay timed regular language $\mathcal{L} \subseteq (\mathbb{R}_{\geq 0} \times \Sigma)^*$. Denoting by $\mathcal{N}_1 \subseteq (\mathbb{R}_{\geq 0} \times \Sigma)^*$ the delay timed regular language introduced in the proof of the previous proposition, we define $\mathcal{V} \subseteq (\mathbb{R}_{\geq 0} \times \Sigma_+)^*$ as the union of the following three delay timed languages: (this is a natural adaptation of *Compose* to delay timed words)

$$\begin{aligned}
\mathcal{V}_1 &= \{w \mid \exists w' \in \mathcal{L},\ \exists w'' \in (\mathbb{R}_{\geq 0} \times \Sigma)^*, \exists \tau \text{ s.t. } w = w' \cdot (c, \tau) \cdot w''\} \\
\mathcal{V}_2 &= \{w \mid |w|_c \neq 1\} \\
\mathcal{V}_3 &= \{w \mid \exists w' \in (\mathbb{R}_{\geq 0} \times \Sigma)^*,\ \exists w'' \in \mathcal{N}_1, \exists \tau \text{ s.t. } w = w' \cdot (c, \tau) \cdot w''\}
\end{aligned}$$

Since $\mathcal{L}$ and $\mathcal{N}_1$ are delay timed regular, we get that $\mathcal{V}$ is also delay timed regular. We consider now the delay timed language $\mathcal{W} = \mathcal{V} \sqcup\!\sqcup\, \mathcal{N}_2$ where $\mathcal{N}_2$ has been defined in the previous proof. Note that $\mathcal{N}_2$ involves letter b. We claim that $\mathcal{L}$ is universal (on Σ) iff $\mathcal{W}$ is ε-delay timed regular. We distinguish two cases:

1. **First case.** We assume $\mathcal{L}$ is universal on Σ, *i.e.*, $\mathcal{L} = (\mathbb{R}_{\geq 0} \times \Sigma)^*$. Then, $\mathcal{V} = (\mathbb{R}_{\geq 0} \times \Sigma_+)^*$, *i.e.*, $\mathcal{V}$ is universal on Σ_+. It is then easy to verify that the TA depicted on Figure 6 recognizes $\mathcal{W}$. In particular, $\mathcal{W}$ is (ε)-delay timed regular.

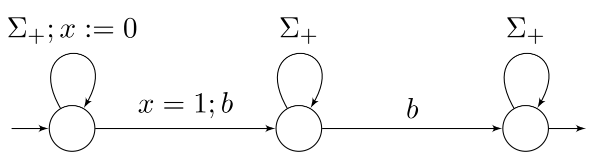

Figure 6. A TA accepting $\mathcal{W}$.

2. **Second case.** We assume $\mathcal{L}$ is not universal on Σ. Towards a contradiction, assume that $\mathcal{W}$ is ε-delay timed regular. Then, the delay timed language $\mathcal{X} = \mathcal{W} \cap ((\mathbb{R}_{\geq 0} \times \Sigma)^* \cdot (1, c) \cdot \mathcal{N}_3)$ is ε-delay timed regular. Pick a delay timed word $w = (\tau_1, a_1) \cdots (\tau_k, a_k) \in (\mathbb{R}_{\geq 0} \times \Sigma)^*$ which does not belong to $\mathcal{L}$. Consider now a delay timed word $x \in (\mathbb{R}_{\geq 0} \times \Sigma_b)^*$. We will show the following equivalence:

$$w \cdot (1, c) \cdot x \in \mathcal{X} \iff x \in (\mathcal{N}_1 \sqcup\!\sqcup\, \mathcal{N}_2) \cap \mathcal{N}_3$$

First suppose that $w' = w \cdot (1, c) \cdot x \in \mathcal{X}$. Since, $w' \in (\mathbb{R}_{\geq 0} \times \Sigma)^* \cdot (1, c) \cdot \mathcal{N}_3$, we get that $x \in \mathcal{N}_3$. Since there is a single occurrence of c in w', w' belongs to either $\mathcal{V}_1 \sqcup\!\sqcup\, \mathcal{N}_2$ or

[4]Just notice that the proof presented in [18] is not completely correct, but it can be fixed using our techniques.

$\mathcal{V}_3 \sqcup\!\sqcup \mathcal{N}_2$. Assume that $w' \in \mathcal{V}_1 \sqcup\!\sqcup \mathcal{N}_2$, thus $w' \in w^- \cdot (1,c) \cdot w^+ \sqcup\!\sqcup w_2$ with $w^- \in \mathcal{L}$ and $w_2 \in \mathcal{N}_2$. Thus $w^- \neq w$ and so w is obtained by inserting letter occurrences of w_2 in w^- but these are b occurrences which cannot occur in w a word over Σ. Hence we have that $w \cdot (1,c) \cdot x \in ((\mathbb{R}_{\geq 0} \times \Sigma)^* \cdot (1,c) \cdot \mathcal{N}_1) \sqcup\!\sqcup \mathcal{N}_2$. Again since a word of $\mathcal{N}_2$ includes only b occurrences, we get $x \in \mathcal{N}_1 \sqcup\!\sqcup \mathcal{N}_2$, which concludes the proof of the first direction. Conversely, the second implication follows from $w \cdot (1,c) \cdot (\mathcal{N}_1 \sqcup\!\sqcup \mathcal{N}_2) \subseteq (w \cdot (1,c) \cdot \mathcal{N}_1) \sqcup\!\sqcup \mathcal{N}_2$.

Then we mimic the proof of Proposition 6.1 and prove that $\mathcal{X}$ cannot be ε-timed regular. However, this is not direct, and requires to be careful. Let denote by $\mathcal{A}$ a TA_ε accepting $\mathcal{X}$. We denote by d its granularity. Consider a delay timed word x belonging to $(\mathcal{N}_1 \sqcup\!\sqcup \mathcal{N}_2) \cap \mathcal{N}_3$ such that:

$$\begin{cases} t_2 & \not\equiv 0 \mod 1/d \\ s + t_2 & \not\equiv 0 \mod 1/d \\ s & \not\equiv 0 \mod 1/d \\ s + \sum_{j=i}^{k} \tau_j & \not\equiv 0 \mod 1/d, \ \forall i \in \{1, \ldots, k\} \end{cases}$$

This is possible since the set of pairs (s, t_2) that do not fulfill one of these equations has zero measure w.r.t. Lebesgue measure. We can then consider the delay timed word $w' = w \cdot (1,c) \cdot x \in \mathcal{X}$. Using the same techniques as in the previous proof, we can exhibit a TA $\mathcal{A}''$ whose granularity divides d and such that $w' \in L(\mathcal{A}'') \subseteq L(\mathcal{A})$. We give the delay timed word w':

$$w' = (\tau_1, a_1) \cdots (\tau_k, a_k) \cdot (1,c) \cdot (t_1, a) \cdot (1,b) \cdot (s,b) \cdot (1,a) \cdot (t_2, a).$$

A simple examination yields that the possible clock values reached immediately before the firing of the last a are the following ones: $t_2, t_2+1, t_2+1+s, t_2+2+s, 3+s, 4+s, 4+s+\tau_k, \ldots, 4+s+\sum_{j=1}^{k} \tau_j$. As a consequence, due to the constraints imposed on x, the region reached at this instant is time-open, and we can postpone the firing of the last a. We obtain another timed word w'' which is accepted by $\mathcal{A}''$, but does not belong to $\mathcal{X}$, since it violates the property $t_1 + t_2 = 1$ required by $(\mathcal{N}_1 \sqcup\!\sqcup \mathcal{N}_2) \cap \mathcal{N}_3$. This yields the contradiction.

This concludes the proof: determining whether $\mathcal{W}$ can be recognized by a TA_ε is equivalent to deciding whether $\mathcal{L}$ is universal. □

7. Extension to Infinite Timed Words

In this section, we explain how all previous results extend to the framework of infinite timed words. First, we define the acceptance of infinite timed words by timed automata with or without silent transitions. We assume that the acceptance condition is given by a Büchi condition, and replace the set of accepting locations F in the definition of a timed automaton by a set of *repeated locations* R. Take $\mathcal{A} = (L, \ell_0, X, E, R)$ such a timed automaton. For defining its semantics in terms of infinite timed words, we need to distinguish between automata with or without silent transitions. We first assume that $\mathcal{A}$ has no silent transitions. Given a infinite timed execution $\varrho : (\ell_0, v_0) \xrightarrow{d_0} (\ell_0, v_0 + d_0) \xrightarrow{a_0} (\ell_1, v_1) \xrightarrow{d_1} (\ell_1, v_1 + d_1) \xrightarrow{a_1} \ldots$, its label is the infinite timed word $w = (a_0, \tau_i)_{i \geq 0}$ where τ_i is given as previously by $\tau_i = \sum_{k \leq i} d_k$. If the timed execution passes infinitely often through a location of R, we say that it is an *accepting execution*, and that its label is accepted by the timed automaton $\mathcal{A}$. Then, we assume that $\mathcal{A}$ is a timed automaton over Σ_ε (that is, it has silent transitions). As in the case of finite timed words, we define by w' the timed word obtained from w by deleting the pairs whose first component is equal to ε. It may be the case that w' is finite: it happens exactly when there are infinitely many actions labelled by ε, but only finitely many labelled by elements different from ε. If the timed execution passes infinitely often through a repeated location, and if moreover w' is *infinite*, we say that ϱ is an accepting execution, and that its label w' is accepted by $\mathcal{A}$. In both cases, the set of infinite timed words accepted by $\mathcal{A}$ is denoted $L^\omega(\mathcal{A})$.

The decidability of the universality problem is different in the case of finite and infinite words w.r.t. the number of clocks. The next theorem which has been established very recently states the case of infinite words.

Theorem 7.1. (Universality problem)
The universality problem over infinite words for TA with a single clock is undecidable [1].

All the results we have presented in the framework of languages of finite timed words extend to the framework of languages of infinite timed words (with a slight modification due to the previous theorem). We sum up all results in the following theorem.

Theorem 7.2. (Infinite words)
The five following problems are undecidable:

1. Given a TA_ε $\mathcal{A}$, determine whether there exists a TA $\mathcal{B}$ such that $L^\omega(\mathcal{B}) = L^\omega(\mathcal{A})$.
2. Given a TA_ε $\mathcal{A}$, determine whether there exists a deterministic TA $\mathcal{B}$ such that $L^\omega(\mathcal{B}) = L^\omega(\mathcal{A})$.
3. Given a TA_ε $\mathcal{A}$ over an alphabet of at least two letters, determine whether there exists a TA_ε $\mathcal{B}$ such that $L^\omega(\mathcal{B}) = \overline{L^\omega(\mathcal{A})}$.
4. Given a TA $\mathcal{A}$ with n clocks ($n \geq 1$), determine whether there exists a TA_ε $\mathcal{B}$ with $n-1$ clocks such that $L^\omega(\mathcal{B}) = L^\omega(\mathcal{A})$.
5. Given two TA $\mathcal{A}$ and $\mathcal{B}$, determine whether the shuffle of $L^\omega(\mathcal{A})$ and $L^\omega(\mathcal{B})$ is ε-timed regular[5].

The proof of this theorem can be derived from the various proofs we have proposed in the framework of finite timed words. The idea is to modify the construction *Compose* for the framework of infinite timed words, and then to build a regular timed language $\mathcal{R}$ (over infinite words) witnessing the strict inclusion between the two families of studied languages.

As previously, given an alphabet Σ, we pick a letter c not in Σ, and denote by Σ_+ the alphabet $\Sigma \cup \{c\}$.

Definition 7.1. Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ and $\mathcal{R} \subseteq \mathcal{TW}^\omega(\Sigma)$ be two timed languages over Σ (the first one only contains finite words, whereas the second one only contains infinite words). Then *Inf-Compose*$(\mathcal{L}, \mathcal{R})$ is a timed language of infinite words over Σ_+ defined as the union of the following three languages:

$$\begin{aligned}
\mathcal{V}_1 &= \{w \in \mathcal{TW}^\omega(\Sigma_+) \mid \exists w' \in \mathcal{L},\ \exists w'' \in \mathcal{TW}^\omega(\Sigma), \exists \tau \text{ s.t. } w = w'(c,\tau)w''\} \\
\mathcal{V}_2 &= \{w \in \mathcal{TW}^\omega(\Sigma_+) \mid |w|_c \neq 1\} \\
\mathcal{V}_3 &= \{w \in \mathcal{TW}^\omega(\Sigma_+) \mid \exists w' \in \mathcal{TW}^*(\Sigma),\ \exists w'' \in \mathcal{R}, \exists \tau \text{ s.t. } w = w'(c,\tau)(w''+\tau)\}
\end{aligned}$$

[5]For this result, we exclude Zeno timed words since the construction of [9] is only valid for infinite non Zeno words.

We obtain similar properties for this new construction:

Lemma 7.1. Let $\mathcal{L} \subseteq \mathcal{TW}^*(\Sigma)$ and $\mathcal{R} \subseteq \mathcal{TW}^\omega(\Sigma)$ be two timed languages over alphabet Σ.

- *Inf-Compose*$(\mathcal{TW}^*(\Sigma), \mathcal{R}) = \mathcal{TW}^\omega(\Sigma_+)$, it is thus accepted by a deterministic TA with no clock.
- If $\mathcal{L}$ and $\mathcal{R}$ are accepted by TA_ε with at most n clocks, then *Inf-Compose*$(\mathcal{L}, \mathcal{R})$ is also accepted by a TA_ε with at most n clocks.

The proof of this lemma is similar to that of Lemma 3.1.

Proof:
[of Theorem 7.2] We detail the main elements of the proof for each item of the theorem. For all of them (except items 2.), the proof proceeds by adapting the definition of the witness language to the context of infinite words, and then applying the reasonning of the case of finite timed words. Indeed, the construction in the case of infinite timed words still considers a language $\mathcal{L}$ which is composed of finite timed words. As a consequence, when considering the case of a non-universal language, one can consider a finite timed word not element of $\mathcal{L}$. It is then routine to check that all the details of the proofs are preserved in this context.

1. *Removing silent transitions.* We define the interpretation of the language $\mathcal{R}_{\mathit{even}}$ in the context of infinite timed words as follows:

$$\mathcal{R}^\omega_{\mathit{even}} = \{(a, \tau_i)_{i\geq 0} \in \mathcal{TW}^\omega(\Sigma) \mid \tau_i \equiv 0 \mod 2 \text{ for every } i \geq 0\}$$

2. *Determinization.* The result is a corollary of Theorem 9 in [18] establishing the undecidability of determinizability for timed Büchi automata.

3. *Complementation over an alphabet with at least two letters.* We define the interpretation of the language $\mathcal{R}_{a,b}$ in the context of infinite timed words as follows:

$$\mathcal{R}^\omega_{a,b} = \{(a_i, \tau_i)_{i\geq 0} \in \mathcal{TW}^\omega(\Sigma) \mid \exists i, a_i = a, \text{ and } \forall j \geq i, \tau_j - \tau_i \neq 1\}$$

   Note that to obtain the undecidability result, it is necessary to adapt the proof by considering the language $\mathcal{T}_1^\omega$ defined as the set of infinite timed words w such that:
   $[(i)]$

   (a) *Untimed*(w) belongs to the untimed regular language $a^*b^+a^\omega$,

   (b) all a's before the first b occur within $[0, 1[$, and

   (c) no two a's in the initial fragment occur at the same date.

4. *Minimization of the number of clocks, case $n \geq 1$.* The interpretation of the language $\mathcal{R}_n$ in the context of infinite timed words is defined as follows:

$$\mathcal{R}^\omega_n = \{(a, \tau_i)_{i\geq 0} \in \mathcal{TW}^\omega(\Sigma) \mid \forall 1 \leq i \leq n,\ 0 \leq \tau_i < 1 \wedge \tau_{n+i} = 1 + \tau_i\}$$

5. *Shuffle of timed languages.* First, to adapt the proof of Proposition 6.1, we modify the definitions of the languages $\mathcal{N}_1, \mathcal{N}_2$ and $\mathcal{N}_3$ as follows:

$$\begin{aligned}
\mathcal{N}_1^\omega &= \{(t_1, a) \cdot (1, a) \cdot (t_2, a) \cdot (1, b)^\omega \mid t_1 + t_2 = 1\} \\
\mathcal{N}_2^\omega &= \{(1, b) \cdot (s, b) \cdot (1, b)^\omega \mid s \in \mathbb{R}_{\geq 0}\} \\
\mathcal{N}_3^\omega &= \{(t_1, a) \cdot (1, b) \cdot (s, b) \cdot (1, a) \cdot (t_2, a) \cdot (1, b)^\omega \mid t_1, s, t_2 \in \mathbb{R}_{\geq 0}\}
\end{aligned}$$

Second, to adapt the proof of Theorem 6.1, we consider a delay timed regular language $\mathcal{L} \subseteq (\mathbb{R}_{\geq 0} \times \Sigma)^*$ and extend the definition of the language $\mathcal{V}$ in the context of infinite words by adapting the construction *Inf-Compose*. $\mathcal{V}^\omega$ is the subset of $(\mathbb{R}_{\geq 0} \times \Sigma_+)^\omega$ defined as the union of the following three languages :

$$\begin{aligned}
\mathcal{V}_1^\omega &= \{w \mid \exists w' \in \mathcal{L},\ \exists w'' \in (\mathbb{R}_{\geq 0} \times \Sigma)^\omega, \exists \tau \text{ s.t. } w = w' \cdot (c, \tau) \cdot w''\} \\
\mathcal{V}_2^\omega &= \{w \mid |w|_c \neq 1\} \\
\mathcal{V}_3^\omega &= \{w \mid \exists w' \in (\mathbb{R}_{\geq 0} \times \Sigma)^*,\ \exists w'' \in \mathcal{N}_1^\omega, \exists \tau \text{ s.t. } w = w' \cdot (c, \tau) \cdot w''\}
\end{aligned}$$

Then, for each of the above results, it is possible to verify that all the details of the proof presented in the case of finite (delay) timed words extend to the case of infinite (delay) timed words.

For some of the points of the previous theorem, another proof would have been possible by "suffixing" the languages by $\mathcal{TW}^\omega(\Sigma)$ (with a natural meaning for the suffixing operation). However observe that, for instance, in the case of complementation and even in the untimed framework (when $\mathcal{L}$ is a language of finite words): $(\mathcal{L}\Sigma^\omega)^c \neq \mathcal{L}^c\Sigma^\omega$. □

8. Conclusion

In this work, we have studied decision problems related to timed automata with silent transitions. We have first answered negatively a central question raised by the introduction of silent transitions: can we decide whether the language recognized by a timed automaton with silent transitions is recognized by some classical timed automaton? Then we have extended undecidability results known in the framework of timed automata. Proofs of these results are more involved than the previous ones because a timed word can be accepted in uncountably many different ways by a timed automaton with silent transitions. In addition to the interest of the results, we believe that such proofs give more insight on the role of silent transitions.

Finally, since all our proofs rely on the introduction of a new letter, a possible future work is the particular case of an alphabet reduced to a single letter.

References

[1] Abdulla, P. A., Deneux, J., Ouaknine, J., Quaas, K., Worrell, J. B.: Universality Analysis for One-Clock Timed Automata, *Fundamenta Informaticae*, **89**(4), 2008.

[2] Adams, S., Ouaknine, J., Worrell, J.: Undecidability of Universality for Timed Automata with Minimal Resources, *Proc. 5th International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS'07)*, 4763, Springer, 2007.

[3] Alur, R., Dill, D. L.: Automata for Modeling Real-Time Systems, *Proc. 17th International Colloquium on Automata, Languages and Programming (ICALP'90)*, 443, Springer, 1990.

[4] Alur, R., Dill, D. L.: A Theory of Timed Automata, *Theoretical Computer Science*, **126**(2), 1994, 183–235.

[5] Alur, R., Fix, L., Henzinger, Th. A.: A Determinizable Class of Timed Automata, *Proc. 6th International Conference on Computer Aided Verification (CAV'94)*, 818, Springer, 1994.

[6] Alur, R., Madhusudan, P.: Decision Problems for Timed Automata: A Survey, *Proc. 4th International School on Formal Methods for the Design of Computer, Communication and Software Systems: Real Time (SFM-04:RT)*, 3185, Springer, 2004.

[7] Asarin, E.: Challenges in Timed Languages: From Applied Theory to Basic Theory, *The Bulletin of the European Association for Theoretical Computer Science*, (83), 2004.

[8] Asarin, E., Caspi, P., Maler, O.: Timed Regular Expressions, *Journal of the ACM*, **49**(2), 2002, 172–206.

[9] Bérard, B., Diekert, V., Gastin, P., Petit, A.: Characterization of the Expressive Power of Silent Transitions in Timed Automata, *Fundamenta Informaticae*, **36**(2–3), 1998, 145–182.

[10] Bouyer, P.: A Logical Characterization of Data Languages, *Information Processing Letters*, **84**(2), 2002, 75–85.

[11] Bouyer, P., Petit, A.: A Kleene/Büchi-like Theorem for Clock Languages, *Journal of Automata, Languages and Combinatorics*, **7**(2), 2002, 167–186.

[12] Bouyer, P., Petit, A., Thérien, D.: An Algebraic Approach to Data Languages and Timed Languages, *Information and Computation*, 2002.

[13] Chevalier, F., D'Souza, D., Prabakhar, P.: On Continuous Timed Automata with Input-Determined Guards, *Proc. 26th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'06)*, 4337, Springer, 2006.

[14] Dima, C.: Kleene Theorems for Event-Clock Automata, *Proc. 12th International Symposium on Fundamentals of Computation Theory (FCT'99)*, 1684, Springer, 1999.

[15] Dima, C.: Real-Time Automata, *Journal of Automata, Languages and Combinatorics*, **6**(1), 2001, 3–24.

[16] Dima, C.: Timed Shuffle Expressions, *Proc. 16th International Conference on Concurrency Theory (CONCUR'05)*, 3653, Springer, 2005.

[17] D'Souza, D.: A Logical Characterization of Event Recording Automata, *Proc. 6th International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems (FTRTFT'00)*, 1926, Springer, 2000.

[18] Finkel, O.: Undecidable Problems About Timed Automata, *Proc. 4th International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS'06)*, 4202, Springer, 2006.

[19] Henzinger, Th. A., Kopke, P. W., Wong-Toi, H.: The Expressive Power of Clocks, *Proc. 22nd International Colloquium on Automata, Languages and Programming (ICALP'95)*, 944, Springer, 1995.

[20] Lasota, S., Walukiewicz, I.: Alternating Timed Automata, *ACM Transactions on Computational Logic*, **9**(2:10), 2008.

[21] Maler, O., Pnueli, A.: On Recognizable Timed Languages, *Proc. 7th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS'04)*, 2987, Springer, 2004.

[22] Ouaknine, J., Worrell, J.: On the Language Inclusion Problem for Timed Automata: Closing a Decidability Gap, *Proc. 19th Annual Symposium on Logic in Computer Science (LICS'04)*, IEEE Computer Society Press, 2004.

[23] Tripakis, S.: Folk Theorems on the Determinization and Minimization of Timed Automata, *Proc. 1st International Workshop on Formal Modeling and Analysis of Timed Systems (FORMATS'03)*, 2791, Springer, 2003.

[24] Wilke, Th.: Specifying Timed State Sequences in Powerful Decidable Logics and Timed Automata, *Proc. 3rd International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems (FTRTFT'94)*, 863, Springer, 1994.