On Timed Regular Languages Defined By Inverse Monoid Morphisms

Cătălin Dima

Laboratoire d'Algorithmique, Complexité et Logique,
Université Paris XII – Val de Marne, 61 av. du Général de Gaulle, 94010 Créteil Cedex, France

Abstract. We prove that the class of timed languages accepted by finite automata, in the sense of Eilenberg (in his book *Automata, Languages, Machines* [Eil74]), does not include any language with "quantitative" timing. That is, once a signal is accepted by a finite automaton, any other signal with the same "untimed trace" is accepted too. The proof makes use of the notion of automaton over a monoid, and is based upon a lemma showing that any nontrivial morphism from the monoid of nonnegative real numbers to a finite monoid maps all positive reals into a "zero" of the finite monoid.

1 Introduction

The automaton is one of the central notions in computer science, as a large part of formal methods is based upon the theory of automata and its connection to different types of logics.

In the past fifteen years there has been considerable interest in generalizing the theory of finite automata to models of real-time systems. The effort has produced many important results, grouped in what now we can call as the *theory of timed automata* [AD94]. These results give a comprehensive extension of the notion of regular languages to the case of dense time domains and of its relations to logical approaches like temporal logics [HRS98] or duration calculus [ZCHR91].

However this generalization has revealed some differences to the classical theory, namely that, unlike emptiness, universality is undecidable for timed automata [AD94], or that regular expressions need intersection and renaming in order to be equivalent to timed automata [ACM02,Her99]. Hence, some alternative models have been developed [HRS98,Dim01b], following different ideas of what regular timed languages should look like, leading to models that less expressive that timed automata.

This paper aims at giving a direct (and negative) answer to the problem of finding the "right" class of "regular timed languages", starting only from considerations from classical automata theory. Our property says, in particular, that real-time devices like clocks in timed automata cannot be simulated by any state-based technique in classical finite automata theory.

We prove here that any timed language (i.e. set of signals) that can be defined as the inverse image of a subset of a finite monoid, does not carry any timing information. That is, whenever a signal σ is in the language, any other signal with the same sequence of states (and with any other durations of these states) is in the language too. This property is based upon a lemma stating that there are at most two morphisms from the monoid of nonnegative reals to any finite monoid: the trivial morphism and, in the eventuality the target monoid has a "zero", the morphism which takes any positive number to this "zero". Hence the problem is traced to the "stuttering" structure of the real numbers.

This result is proved using a "diagram-chasing" technique specific to category theory [Mac71]. This proof takes advantage of the algebraic presentation of signals and we believe it is drastically shorter than any other proof, that would need to mimic the uniqueness properties of the monoid of signals.

The paper is organized as follows: in the second section we recall the coproduct representation of signals and the fact that, similarly to languages of words, the powerset of signals can be organized as a Kleene algebra. We also remind in this section the notion of finite automata over a monoid, from [Eil74]. The third section is concerned with the negative result concerning the "lack of quantitative timing" of timed languages defined by inverse monoid morphisms. We end with a section with conclusions and relation to other work.

2 Signals – basic properties

This section recalls some standard basic definitions concerning one of the semantical models of timed systems : the *signals* (see e.g. [ACM97]). Some of them (viz. the notions of length and untiming) are restated in algebraic form due to the algebraic form of the proofs in this paper. We also remind the notion of automaton over a finite monoid from [Eil74] at the end of this section.

We denote $\mathbb{R}_{\leq 0}$, $\mathbb{R}_{\geq 0}$ and $\mathbb{R}_{>0}$ the sets of negative, nonnegative, resp. positive numbers, $\mathbb{Q}_{\geq 0}$ the set of nonnegative rational numbers, $\mathbb{Z}$ the set of integers and $\mathbb{N}$ the set of nonnegative integers (also called *naturals*). For each $n_1, n_2 \in \mathbb{Z}$, $[n_1 \ldots n_2]$ denotes the *interval of integers* $\{n_1, n_1 + 1, \ldots, n_2\}$, while $]2, 3]$ denotes the left-open, right-closed interval of reals whose limits are 2, resp. 3. $\mathbb{Q}Int$ denotes the set of intervals having bounds in $\mathbb{Q}_{\geq 0} \cup \{\infty\}$ and including the empty set, while $\mathbb{Z}Int$ denotes the set of intervals having bounds in $\mathbb{Z} \cup \{-\infty, \infty\}$ and including the empty set. An open interval is denoted as $]a, b[$, while a closed one is denoted as $[a, b]$. We will extensively use left-closed right-open intervals, which are thence denoted $[a, b[$.

For each function $f : \mathbb{R} \to A$, real number α and each $a \in A$, we say that *the left limit of f at α is a* and denote it $\lim\limits_{t \nearrow \alpha} f = a$ iff the following property holds:

$$\text{there exists some } \varepsilon > 0 \text{ such that for all } t \in]\alpha - \varepsilon, \alpha[, f(t) = a$$

Right limits $\lim\limits_{t \searrow \alpha} f = a$ can be defined similarly. This definition amounts to considering that the set A is equipped with the *discrete topology*.

A *left discontinuity* in f is some $\alpha \in \mathbb{R}$ for which $\lim\limits_{t \nearrow \alpha} f(t) \neq f(\alpha) = \lim\limits_{t \searrow \alpha} f(t)$. The discontinuity is *right* if we rather have that $\lim\limits_{t \nearrow \alpha} f(t) = f(\alpha) \neq \lim\limits_{t \searrow \alpha} f(t)$.

Definition 1. *A* ***signal*** *over a finite alphabet Σ is a function $\sigma : [0, \alpha[\to \Sigma$ where α is a nonnegative number, function which has finitely many discontinuities, all of them being left discontinuities.*

We denote $dom(\sigma)$ the domain of σ and $\ell(\sigma)$ its endpoint. $\ell(\sigma)$ is also called the *length* of σ.

Signals can be given graphical representation. For example, Figure 1 gives the graphical representation of the signal $\sigma_1 : [0, 6[\to \{a, b, c\}$ defined by:

$$\sigma_1(t) = \begin{cases} a & \text{iff } t \in [0, 1[\cup[\pi, 4.25[\\ b & \text{iff } t \in [1, \pi[\\ c & \text{iff } t \in [4.25, 6[\end{cases} \tag{1}$$

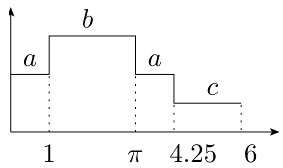

Fig. 1. A graphical representation of the signal defined in Identity 1.

$\mathsf{Sig}(\Sigma)$ denotes the set of signals over Σ. Note that there exists a unique signal with empty domain $\sigma_\epsilon : [0, 0[\to \Sigma$.

For $\sigma_1, \sigma_2 \in \mathsf{Sig}(\Sigma)$ with $dom(\sigma_i) = [0, e_i[$ $(i = 1, 2)$ define their *concatenation* $\sigma_1 \cdot \sigma_2 = \sigma$ as the signal with $dom(\sigma) = [0, e_1 + e_2[$ and such that

$$\sigma(t) = \begin{cases} \sigma_1(t) & \text{for } t \in [0, e_1[, \\ \sigma_2(t - e_1) & \text{for } t \in [e_1, e_1 + e_2[. \end{cases}$$

For example, the signal σ_1 in Figure 1 can be regarded as the concatenation of the two signals in Figure 2.

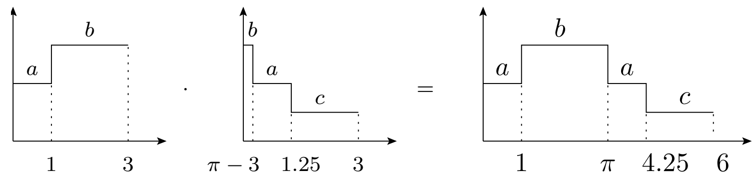

Fig. 2. An example of signal concatenation.

Before going into the details of the proof of our result, we remind briefly the definition of coproducts of monoids:

Definition 2. *The* ***coproduct*** *of a family of monoids* $(M_i, \cdot, e_i)_{i \in \mathcal{I}}$ *is the monoid* $(M, \cdot, e)$ *defined as the quotient:* $M = \left(\coprod_{i \in \mathcal{I}} M_i \right)^* \Big/ _\sim$ *where*

- $\coprod$ *is the usual disjoint sum:* $\coprod_{i \in \mathcal{I}} M_i = \{(m_i, i) \mid i \in \mathcal{I}, m_i \in M_i\}$*;*

– $\left(\coprod_{i\in\mathcal{I}} M_i\right)^*$ *is the free monoid over* $\coprod_{i\in\mathcal{I}} M_i$ *(with concatenation denoted as juxtaposition and empty sequence as* ε*);*
– *and* $\sim$ *is the congruence on* $\left(\coprod_{i\in\mathcal{I}} M_i\right)^*$ *generated by the equations:*

$$(m_i, i)(m_i', i) \sim (m_i \cdot m_i', i) \qquad \forall m_i, m_i' \in M_i, i \in \mathcal{I} \tag{2}$$
$$(e_i, i) \sim (e_j, j) \qquad \forall i, j \in \mathcal{I} \tag{3}$$
$$(e_i, i) \sim \varepsilon \qquad \forall i, \in \mathcal{I} \tag{4}$$

We denote the coproduct of the family $(M_i, \cdot, e_i)_{i\in\mathcal{I}}$ as $\bigoplus_{i\in\mathcal{I}} M_i$. Note that the unit of $\bigoplus_{i\in\mathcal{I}} M_i$ is the class of ε. we denote this unit as e.

Observe that each element $\mathbf{m} \in \bigoplus_{i\in\mathcal{I}} M_i$ can be uniquely represented as a finite concatenation of "atomic" elements

$$\mathbf{m} = [m_{i_1}, i_1] \cdot \ldots \cdot [m_{i_k}, i_k] \tag{5}$$

in which $[m_{i_j}, i_k] \neq e$ for all $j \in [1 \ldots k]$.

Theorem 1. *The monoid* $\bigoplus_{i\in\mathcal{I}} M_i$ *has the following* universality property*: for any monoid* $(M, \cdot, e)$ *and family of morphisms* $\phi_i : M_i \to M$*, there exists a unique morphism* $\phi : \bigoplus_{i\in\mathcal{I}} M_i \to M$ *such that* $\phi([m_i, i]) = \phi_i(m_i)$*. This morphism is denoted* $\langle\phi_i\rangle_{i\in\mathcal{I}}$ *and is called the* ***coproduct*** *of the family* $(\phi_i)_{i\in\mathcal{I}}$.

This theorem is depicted in Figure 3. Here, ι_i denotes the inclusion morphism.

$$\begin{tikzcd} M_i \arrow[r, hook, "\iota_i"] \arrow[dr, "\phi_i"'] & \bigoplus_{i\in\mathcal{I}} M_i \arrow[d, "\exists!\langle\phi_i\rangle_{i\in\mathcal{I}}"] \\ & M \end{tikzcd}$$

Fig. 3. The commutative diagram for Theorem 1.

The construction of the coproduct morphism is the following: each element $\mathbf{m} \in \bigoplus_{i\in\mathcal{I}} M_i$ is decomposed as in Identity 5, $\mathbf{m} = [m_{i_1}, i_1] \cdot \ldots [m_{i_k}, i_k]$. We then put:

$$\langle\phi_i\rangle_{i\in\mathcal{I}}(m) = \phi_{i_1}(m_{i_1}) \cdot \ldots \phi_{i_k}(m_{i_k})$$

When the family $\mathcal{I}$ is finite, say $\mathcal{I} = \{1, \ldots, n\}$, we denote the coproduct as $\bigoplus_{i=1}^n M_i$, and when the monoids M_i are identical (this implies that their operations and units are the same too) we denote it as $\bigoplus_{i\in\mathcal{I}} M$. Finally, when $\mathcal{I} = \{1, \ldots, n\}$ and $M = M_i$ for all $i \in \mathcal{I}$, we denote the coproduct as $\bigoplus_{i=1}^n M$.

Proposition 1 ([ACM02]). $(\mathsf{Sig}(\Sigma), \cdot, \sigma_\epsilon)$ *is a a noncommutative monoid, called the* ***monoid of signals with concatenation****.*

Moreover $(\mathsf{Sig}(\Sigma), \cdot, \sigma_\epsilon)$ *is isomorphic to the coproduct* $\bigoplus_{a\in\Sigma} \mathbb{R}_{\geq 0}$ *of* $card(\Sigma)$ *copies of the monoid of nonnegative reals* $(\mathbb{R}_{\geq 0}, +, 0)$*.*

Proof. It is clear that the domain of each signal σ splits into finitely many intervals $[e_{i-1}, e_i[$ ($i \in [1 \ldots n]$) on which σ is constant. Therefore we may identify σ with the formal concatenation:

$$\phi(\sigma) = a_1^{t_1} a_2^{t_2} \ldots a_n^{t_n} \text{ where } t_i = e_i - e_{i-1} \text{ and } a_i = \sigma(t)\ \forall t \in [e_{i-1}, e_i[\tag{6}$$

If we add $\phi(\sigma_\epsilon) = 0$ we obtain the isomorphism $\phi : \mathsf{Sig}(\Sigma) \to \bigoplus_{a\in\Sigma} \mathbb{R}_{\geq 0}$. □

As we can see, this proposition gives a more "friendly" presentation of signals [ACM97]: the signal presented in Figure 1 is represented also by the following element of $\bigoplus_{a\in\Sigma} \mathbb{R}_{\geq 0}$:

$$\sigma_1 = a^1 b^{\pi-1} a^{4.25-\pi} c^{1.75}$$

Observe also that the empty signal σ_ϵ is denoted by the empty sequence ε and that $a^0 = \varepsilon = \sigma_\epsilon$ for any $a \in \Sigma$. We will utilize both notations ε and σ_ϵ for the unit of concatenation in $\mathsf{Sig}(\Sigma)$.

Proposition 1 allows us to define the length of a signal as a monoid morphism induced by the coproduct property: denote first $\iota_a : \mathbb{R}_{\geq 0} \to \mathsf{Sig}(\Sigma)$ the coproduct inclusion for each copy of $\mathbb{R}_{\geq 0}$. Then ℓ is the unique morphism $\ell : \mathsf{Sig}(\Sigma) \to \mathbb{R}_{\geq 0}$ defined by the following diagram:

$$\begin{tikzcd}
\mathbb{R}_{\geq 0} \arrow[r, hook, "\iota_a"] \arrow[dr, "\mathbf{1}_{\mathbb{R}_{\geq 0}}"'] & \bigoplus_{a\in\Sigma}\mathbb{R}_{\geq 0} \arrow[d, "\exists!\ell"] \\
& \mathbb{R}_{\geq 0}
\end{tikzcd}$$

For each symbol $a \in \Sigma$ we denote $\mathsf{Sig}_a(\Sigma)$ the set of signals whose value is constantly a:

$$\mathsf{Sig}_a(\Sigma) := \iota_a(\mathbb{R}_{\geq 0}) = \{a^t \mid t \in \mathbb{R}_{\geq 0}\}$$

Subsets of $\mathsf{Sig}(\Sigma)$ are called **timed languages**. The set of timed languages is denoted $\mathcal{P}(\mathsf{Sig}(\Sigma))$. An example of a timed language is the following:

$$L = \left\{a^{t_1} b^{t_2} \mid t_1 + t_2 \in [1, 3]\right\} \tag{7}$$

that is, the set of signals in which state a is followed by state b, and the overall duration of any signal in L is in the interval $[1, 3]$.

Signal concatenation can be naturally extended to timed languages:

$$L_1 \cdot L_2 = \{\sigma_1 \sigma_2 \mid \sigma_1 \in L_1, \sigma_2 \in L_2\}$$

and gives rise to star:

$$L^* = \bigcup_{n\in\mathbb{N}} L^n$$

where $L^0 = \{\sigma_\epsilon\}$ and $L^{n+1} = L^n \cdot L$.

Proposition 2. $\left(\mathcal{P}(\mathsf{Sig}(\Sigma)), \cup, \cdot, (\cdot)^*, \emptyset, \{\sigma_\epsilon\}\right)$ *is a Kleene algebra.*

We end this section by reminding the way regular languages are related to monoid morphisms in classical automata theory:

Definition 3 ([Eil74]). *Given $(M, \cdot, e)$ a (possibly infinite) monoid, an M-**automaton** is a tuple $\mathcal{A} = (Q, M, \delta, q_0, Q_f)$ where $q_0 \in Q$, $Q_f \subseteq Q$ and $\delta : Q \times M \to Q$ has the property that for all $q \in Q$, $m_1, m_2 \in M$:*

$$\delta(q, m_1 \cdot m_2) = \delta(\delta(q, m_1), m_2) \quad \textit{and} \quad \delta(q, e) = q \tag{8}$$

$\mathcal{A}$ is called **finite** whenever Q is finite. The **accepted language** of $\mathcal{A}$ is $L(\mathcal{A}) = \{m \in M \mid \delta(q_0, m) \in Q_f\}$. It is also assumed that all states $q \in Q$ in an M-automaton are *accessible*, i.e. there exists some $m \in M$ such that $\delta(q_0, m) = q$.

Definition 4. *Given a monoid M, the family of M-**regular languages**, denoted* $\mathsf{Reg}(M)$*, is the family of subsets $L \subseteq M$ which are the accepted language of some* finite *M-automaton.*

Theorem 2 ([Eil74]). $\mathsf{Reg}(M)$ *coincides with the family of subsets $L \subseteq M$ for which there exists some finite monoid $(F, \circ, e')$, some surjective monoid morphism $\phi : M \to F$ and some subset $F' \subseteq F$ such that $L = \phi^{-1}(F')$.*

3 Timed languages and finite automata over timed languages

This section gives our main result on the form of timed languages defined by finite automata. To this end, we introduce the notion of *essentially untimed languages* as a formalization of timed languages that do not carry any quantitative timing information.

Essentially untimed languages are inverse images of classical languages of words, under the *untiming* morphism. The **untiming** is the sequence of symbols that appear in it. Observe that in the untiming of a signal, two consecutive symbols are distinct. Hence the untiming application (actually, morphism) is not surjective.

We will take advantage of the algebraic definition of $\mathsf{Sig}(\Sigma)$ and define the untiming as a coproduct of morphisms. We define first the monoid $SF(\Sigma)$ of **stuttering-free words**, the set of words in which no two consecutive letters are equal. This monoid is endowed with a concatenation operation that "fuses" two identical letters. For example,

$$aba \cdot ac = abac$$

We may define $SF(\Sigma)$ in two ways: as a coproduct monoid and as a quotient monoid:

The *coproduct* definition is the following: for each $a \in \Sigma$ consider the monoid $(S_a = \{\varepsilon, a\}, \cdot, \varepsilon)$ where $a \cdot a = a$ (i.e. a is idempotent). Define then $SF(\Sigma^*)$ as the coproduct of the family monoids $(S_a, \cdot, \varepsilon)_{a \in \Sigma}$,

$$SF(\Sigma) = \bigoplus_{a \in \Sigma} S_a$$

and denote $\kappa_a : S_a \hookrightarrow SF(\Sigma)$ the inclusion morphism which defines this coproduct.

The *quotient* presentation of $SF(\Sigma)$ is the following: consider the relation $\rho \subseteq \Sigma \times \Sigma$ defined by

$$\rho = \{(aa, a) \mid a \in \Sigma\}$$

This relation can be uniquely extended to a congruence on Σ^* as follows:

$$\tilde{\rho} = \{(w_1 aaw_2, w_1 aw_2), (w_1 aw_2, w_1 aaw_2) \mid w_1, w_2 \in \Sigma^*, a \in \Sigma\}$$

Then $SF(\Sigma)$ is isomorphic to the quotient of the Σ^* by the congruence $\tilde{\rho}$. That is, elements of $SF(\Sigma)$ can be thought as equivalence classes w.r.t. $\tilde{\rho}$. We denote $\overbrace{\cdot} : \Sigma^* \to \Sigma^*/\tilde{\rho}$ the canonical projection. For example, $\overbrace{aabcbba} = abcba$.

Concatenation of stuttering-free words, i.e. the *internal* operation on $SF(\Sigma)$ that makes it a monoid, can then be defined as follows: for each $w, w' \in SF(\Sigma)$, w not ending in a and w' not beginning in b,

$$wa \cdot bw' = \begin{cases} wabw' & \text{iff } a \neq b \\ waw' & \text{iff } a = b \end{cases}$$

We may now proceed to the algebraic definition of untiming: consider the following monoid morphism:

$$\rho_a : (\mathbb{R}_{\geq 0}, +, 0) \to (S_a, \cdot, \varepsilon), \quad \rho_a(t) = \begin{cases} a & \text{iff } t \neq 0 \\ \varepsilon & \text{iff } t = 0 \end{cases}$$

The following diagram defines, by theorem 1, the **untiming morphism** $\mathcal{U}$:

$$\begin{tikzcd}
\mathbb{R}_{\geq 0} \arrow[r, hook, "\iota_a"] \arrow[d, "\rho_a"'] & \mathsf{Sig}(\Sigma) = \bigoplus_{a \in \Sigma} \mathbb{R}_{\geq 0} \arrow[d, "\exists! \mathcal{U} = \langle \kappa_a \circ \rho_a \rangle_{a \in \Sigma}"] \\
S_a \arrow[r, hook, "\kappa_a"] & SF(\Sigma)
\end{tikzcd}$$

Definition 5. *A timed language $L \subseteq \mathsf{Sig}(\Sigma)$ is called **essentially untimed** iff there exists some set $L' \subseteq SF(\Sigma)$ (i.e.* a language in the classical sense*) such that $L = \mathcal{U}^{-1}(L')$. The class of **essentially untimed regular languages** consists of essentially untimed languages that are inverse images of regular languages in $SF(\Sigma)$.*

An example of a timed language which is not essentially untimed is the language L seen in 7 above:

$$L = \left\{a^{t_1} b^{t_2} \mid t_1 + t_2 \in [1, 3]\right\}$$

Observe that $SF(\Sigma)$-regular languages in the sense of definition 4 are in fact regular languages in the classical sense [HU92] with the restriction that in each word no two consecutive letters are equal – and this restriction means intersection with a regular set.

3.1 Syntactic monoids on $(\mathsf{Sig}(\Sigma), \cdot, \sigma_\epsilon)$ are not interesting

In this section we show that $\mathsf{Sig}(\Sigma)$-regular languages are essentially untimed. Remind first that a **zero element** in a monoid $(M, \cdot, e)$ is an element $\uparrow \in M$ which satisfies

$$\forall x \in M,\ \uparrow \cdot x = x \cdot \uparrow = \uparrow$$

Our result relies on the following lemma:

Lemma 1. *Suppose $f : (\mathbb{R}_{\geq 0}, +, 0) \to (M, \cdot, e)$ is a surjective monoid morphism and $(M, \cdot, e)$ is a finite monoid. Then*

- *either f is the trivial morphism $f(x) = e$, $\forall x \in \mathbb{R}_{\geq 0}$ (and hence $M = \{e\}$);*
- *or $M = \{e, \uparrow\}$ where $\uparrow$ is a zero element and f maps any $x \neq 0$ to $\uparrow$ and 0 to e.*

Proof. We note first that the surjectivity of f implies that $\cdot$ is commutative. We may also prove by induction on k that for all $x \in \mathbb{R}_{\geq 0}$ and for all $k \in \mathbb{N}$, $\quad f(k \cdot x) = f(x)^k$.

Let's prove first the following:

Claim. If M contains only idempotents then it has at most two elements, and one of these elements is a zero element (in fact *the* zero element, since it is unique).

Proof. Suppose $M = \{m_1, \ldots, m_n\}$. Then $m = m_1 \cdot \ldots \cdot m_n$ is the zero element because

$$\begin{aligned} m \cdot m_i &= m_1 \cdot \ldots \cdot m_n \cdot m_i = m_1 \cdot \ldots \cdot m_i \cdot m_i \cdot \ldots \cdot m_n \\ &= m_1 \cdot \ldots \cdot m_i \cdot \ldots \cdot m_n = m \end{aligned}$$

where we have used the commutativity of $\cdot$.

Two cases arise then: the first is when $m = e$, the unit of M. But the unit can be a zero element only if the monoid has only one element, the unit:

$$e = e \cdot m_i \text{ (as } e \text{ is the zero element) } = m_i \text{ (as } e \text{ is the unit)}$$

The second case is $m \neq e$. In this case there must exist $\beta \neq 0$ such that $f(\beta) = m$.

Take then any $\alpha > 0$. Since $f(\alpha)$ is idempotent we have that $f(\alpha) = f(k \cdot \alpha)$ for any $k \in \mathbb{N}_{\geq 1}$. If we take then $k \geq \lceil \frac{\beta}{\alpha} \rceil$ we obtain $k\alpha \geq \beta$ and hence $k\alpha - \beta \geq 0$, hence $f(k\alpha - \beta)$ is defined.

But then, as $m = f(\beta)$ is the zero element we have that $f(\beta) = f(\beta) \cdot f(k\alpha - \beta)$. Therefore

$$m = f(\beta) = f(\beta) \cdot f(k\alpha - \beta) = f(\beta + k\alpha - \beta) = f(k\alpha) = f(\alpha)$$

which proves that any positive real α is mapped to the zero element. □

So what is left to prove is that if f is surjective then M contains only idempotents. Take a positive real $\alpha \geq 0$ and denote $G(\alpha)$ the image under f of the submonoid $\alpha\mathbb{N}$ of *multiples* of α:

$$G(\alpha) = \big\{ f(n\alpha) \mid n \in \mathbb{N} \big\}$$

It is clear that $G(2^p\alpha) \subseteq G(2^{p+1}\alpha)$ for any $p \in \mathbb{Z}$. But since M is finite, there must exist some $p_0 \in \mathbb{Z}$ such that $G(2^{p_0}\alpha) = G(2^{p_0+1}\alpha)$. Let us then denote $\beta = \min(\alpha, 2^{p_0}\alpha)$, hence $G(\beta) = G\Big(\dfrac{\beta}{2}\Big)$.

We may prove that $G(\beta)$ is *cyclic*, that is, if we denote $k = cardG(\beta)$ then

$$G(\beta) = \big\{f(i\beta) \mid i \in [1 \dots k]\big\} = \big\{f(0), f(\beta), \dots, f(k\beta)\big\} \tag{9}$$

$$f((k+1)\beta) = f(\beta) \tag{10}$$

To this end, note that if there exist $i \in [1 \dots k]$, $j \in \mathbb{N}$ with $i \neq j$ and such that $f(i\beta) = f(j\beta)$ then we would have that $f((i+n)\beta) = f((j+n)\beta)$ for all $n \in \mathbb{N}$. But this would imply that

$$G(\beta) = \big\{f(0), \dots, f((j-1)\beta)\big\}$$

and hence $cardG(\beta) \leq j - 1$, which implies that $j \geq k+1$. This shows that Identity 9 holds.

For showing that Identity 10 holds too, suppose that $f((k+1)\beta) = f(i\beta)$ for some $i \in [1 \dots k]$. By a simple induction argument, we may prove that, for all $n \in \mathbb{N}$,

$$f\big((k+1+n(k+1-i))\beta\big) = f\big((i+n(k+1-i))\beta\big) \tag{11}$$

On the other hand, since $f\Big(\dfrac{\beta}{2}\Big) \in G(\beta)$, we must have some $j \in [0 \dots k]$ such that $f\Big(\dfrac{\beta}{2}\Big) = f(j\beta)$ too. Therefore,

$$f(\beta) = f\Big(\frac{\beta}{2}\Big) \cdot f\Big(\frac{\beta}{2}\Big) = f(j\beta) \cdot f(j\beta) = f(2j\beta)$$

It follows that $2j \geq k+1$. By recursively appying the Identity 11 we further get that:

$$\begin{aligned} f(2j\beta) = f\big((2j-(k+1-i))\beta\big) = f\big((2j-2(k+1-i))\beta\big) = \dots \\ = f\bigg(\Big(2j - \Big(\Big\lfloor\frac{2j-k}{k+1-i}\Big\rfloor + 1\Big)(k+1-i)\Big)\beta\bigg) \end{aligned} \tag{12}$$

and $2j - \Big(\Big\lfloor\dfrac{2j-k}{k+1-i}\Big\rfloor + 1\Big)(k+1-i) \in [i \dots k]$.

But this rewrites to the fact that $f(\beta) = f(l\beta)$ for some $l \in [i \dots k]$, which would be impossible, by Identity 9, unless $i = l = 1$. Hence Identity 10 holds too.

But then, starting from Identity 12 and replacing i with 1 we may further conclude that

$$f(\beta) = f(2j\beta) = f\bigg(\Big(2j - \Big(\Big\lfloor\frac{2j-k}{k}\Big\rfloor + 1\Big) \cdot k\Big)\beta\bigg) = f\bigg(\Big(2j - \Big\lfloor\frac{2j}{k}\Big\rfloor \cdot k\Big)\beta\bigg)$$

with $2j - \Big\lfloor\dfrac{2j}{k}\Big\rfloor k \in [1 \dots k]$. This also implies, by Identity 9, that

$$2j - \Big\lfloor\frac{2j}{k}\Big\rfloor k = 1$$

As a consequence, $k \mid 2j-1$, fact which, corroborated with the hypothesis that $j \in [1 \dots k]$, implies that $k = 2j - 1$.

Therefore,

$$f\left(\frac{\beta}{2}\right) = f((k+1)\beta) = f(\beta) \text{ (by Identity 10)}$$

or, by multiplying by 2,

$$f(\beta) = f(2\beta) = f(\beta)^2.$$

Hence $f(\beta)$ is an idempotent. By an easy induction we may show then that $f(2^p\beta)$ is an idempotent too, which implies that $f(\alpha) = f(2^{p_0}\beta)$ is an idempotent. ☐

Observe that it was essential to utilize the fact that $f\left(\frac{\beta}{2}\right) \in G(\beta)$ to show that $G(\beta)$ is cyclic.

Theorem 3. *The family of* $\mathsf{Sig}(\Sigma)$*-regular languages is equal with the family of essentially untimed regular languages.*

Proof. This is a corollary of theorem 2 and the above lemma. The right-to-left inclusion follows by an easy argument: if K is a regular language in $SF(\Sigma)$, then we have some surjective morphism $\phi : SF(\Sigma) \to M$ where $(M, \cdot, e)$ is a finite monoid, and some subset $F \subseteq M$ such that $K = \phi^{-1}(F)$. But then $\phi \circ \mathcal{U} : \mathsf{Sig}(\Sigma) \to M$ is a surjective monoid morphism too and defining $L = \mathcal{U}^{-1}(K) = (\phi \circ \mathcal{U})^{-1}(F)$ we get that L is a regular language in $(\mathsf{Sig}(\Sigma), \cdot, \sigma_\epsilon)$.

For the reverse inclusion, suppose we have some finite monoid $(M, \cdot, e)$ and some surjective morphism $\phi : \mathsf{Sig}(\Sigma) \to M$. We look for a decomposition of ϕ in which the untimed morphism $\mathcal{U}$ occurs.

Remember that, for any $a \in \Sigma$, $\iota_a : \mathbb{R}_{\geq 0} \to \mathsf{Sig}(\Sigma)$ is the inclusion morphism for the coproduct property. It follows that $\phi \circ \iota_a : \mathbb{R}_{\geq 0} \to M$ is a monoid morphism. This implies, by the above lemma, that the image of this morphism, $Im(\phi \circ \iota_a)$, is a submonoid of M having at most two elements, $M = \{e, \uparrow\}$ with $\uparrow \cdot \uparrow = \uparrow$.

But the monoid $(S_a = \{\varepsilon, a\}, \cdot, \varepsilon)$ which was used to define $\mathcal{U}$ on page 6 is isomorphic to $Im(\phi \circ \iota_a)$, since $aa = a$ in S_a. Define then the morphism $j_a : S_a \to M$ as

$$j_a(\varepsilon) = e \ \text{ and } \ j_a(a) = \phi(\iota_a(1)) = \phi(\iota_a(t)) \text{ for all } t \neq 0$$

That is, j_a is either the isomorphism from S_a to M or the trivial morphism.

Define also the morphisms $\chi_a : \mathbb{R}_{\geq 0} \to S_a$ and $\pi_a : S_a \to S_a$ as follows:

$$\begin{aligned} &\chi_a(0) = \varepsilon \ \text{ and for all } \alpha \neq 0, && \chi_a(t) = \begin{cases} a & \text{iff } \phi(\iota_a(t)) \neq e \\ \varepsilon & \text{otherwise} \end{cases} \\ &\pi_a(\varepsilon) = \varepsilon && \pi_a(a) = \chi_a(1) = \chi_a(t) \text{ for any } t \neq 0 \end{aligned}$$

Then $\phi \circ \iota_a = j_a \circ \chi_a$ and $\chi_a = \pi_a \circ \rho_a$.

Now we are ready to chase the following diagram, in which all the squares and triangles commute:

$$\begin{tikzcd}
\mathbb{R}_{\geq 0} \arrow[r, "\rho_a"] \arrow[d, hook, "\iota_a"'] \arrow[rr, bend left, "\chi_a"] & S_a \arrow[r, "\pi_a"] \arrow[d, hook, "\kappa_a"'] & S_a \arrow[d, hook, "\kappa_a"'] \arrow[dr, "j_a"] & \\
\mathsf{Sig}(\Sigma) \arrow[r, "\exists! \langle \kappa_a \circ \rho_a \rangle_{a \in \Sigma}"] \arrow[rrr, bend right, "\phi"'] & SF(\Sigma) \arrow[r, "\exists! \langle \kappa_a \circ \pi_a \rangle_{a \in \Sigma}"] & SF(\Sigma) \arrow[r, "\exists! \langle j_a \rangle_{a \in \Sigma}"] & M
\end{tikzcd}$$

The upper triangle and the outer square commute as shown above. The inner squares and the right triangle commute by just the coproduct property. The outcome of this diagram is the commutativity of the bottom square, i.e.

$$\phi = \langle j_a \rangle_{a \in \Sigma} \circ \langle \kappa_a \circ \pi_a \rangle_{a \in \Sigma} \circ \langle \kappa_a \circ \rho_a \rangle_{a \in \Sigma} = \langle j_a \rangle_{a \in \Sigma} \circ \langle \kappa_a \circ \pi_a \rangle_{a \in \Sigma} \circ \mathcal{U} \tag{13}$$

since $\mathcal{U} = \langle \kappa_a \circ \rho_a \rangle_{a \in \Sigma}$.

This commutativity follows by the coproduct property: both the left-hand side and the right-hand side, when composed with ι_a, give $j_a \circ \chi_a$. Therefore, both must be equal to the unique coproduct $\langle j_a \circ \chi_a \rangle_{a \in \Sigma}$. Hence we have managed to show that the untiming morphism $\mathcal{U} = \langle \kappa_a \circ \rho_a \rangle_{a \in \Sigma}$ occurs in some decomposition of ϕ.

So suppose we have some $\mathsf{Sig}(\Sigma)$-regular language L, witnessed by the subset $F \subseteq M$, i.e. $L = \phi^{-1}(F)$. Consider then the $SF(\Sigma)$-regular language $K = \big(\langle j_a \rangle_{a \in \Sigma} \circ \langle \kappa_a \circ \pi_a \rangle_{a \in \Sigma}\big)^{-1}(F)$. Then, using the decomposition 13 we get

$$L = \phi^{-1}(F) = \Big(\langle j_a \rangle_{a \in \Sigma} \circ \langle \kappa_a \circ \pi_a \rangle_{a \in \Sigma} \circ \mathcal{U}\Big)^{-1}(F) \; = \mathcal{U}^{-1}(K)$$

Hence L is indeed essentially untimed. $\square$

4 Conclusions

We have proved that a purely algebraical definition of the notion of "regular timed languages" does not capture languages with quantitative timing, as those used in the modeling of real-time systems. As a consequence, more involved automata constructions, like automata with real clocks [AD94] or counters [BPT01] cannot be avoided when modeling such quantitative timing aspects. Note also that [Dim03] suggests that languages accepted by timed automata are closer to a subclass of context-sensitive languages in the Chomsky hierarchy.

References

[ACM97] E. Asarin, P. Caspi, and O. Maler. A Kleene theorem for timed automata. In *Proceedings of LICS'97*, pages 160–171, 1997.

[ACM02] E. Asarin, P. Caspi, and O. Maler. Timed regular expressions. *Journal of ACM*, 49:172–206, 2002.

[AD94] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[BPT01] P. Bouyer, A. Petit, and D. Thérien. An algebraic characterization of data and timed languages. In *Proceedings of CONCUR'2001*, volume 2154 of *LNCS*, pages 248–261, 2001.

[Dim01a] C. Dima. *An algebraic theory of real-time formal languages*. PhD thesis, Université Joseph Fourier Grenoble, France, 2001.

[Dim01b] C. Dima. Real-time automata. *Journal of Automata, Languages and Combinatorics*, 6:3–23, 2001.

[Dim03] C. Dima. A nonarchimedian discretization for timed languages. In *Proceedings of FORMATS'03*, volume 2791 of *LNCS*, pages 161–181, 2003.

[Eil74] S. Eilenberg. *Automata, Languages, and Machines*, volume A. Academic Press, 1974.

[Her99] P. Herrmann. Renaming is necessary in timed regular expressions. In *Proceedings of FST&TCS'99*, volume 1738 of *LNCS*, pages 47–59, 1999.

[HRS98] T. Henzinger, J.-F. Raskin, and P.-Y. Schobbens. The regular real-time languages. In *Proceedings of ICALP'98*, volume 1443 of *LNCS*, pages 580–591, 1998.

[HU92] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley/Narosa Publishing House, 1992.

[Mac71] S. MacLane. *Categories for the Working Mathematician*. Springer Verlag, 1971.

[ZCHR91] Zhou Chaochen, C. A. R. Hoare, and A. P. Ravn. A calculus of durations. *Information Processing Letters*, 40:269–276, 1991.