Refinements and abstractions of signal-event (timed) languages

Béatrice Bérard[1], Paul Gastin[2] and Antoine Petit[2]

[1] LAMSADE, Université Paris Dauphine & CNRS,
Place du Maréchal de Lattre de Tassigny, F-75775 Paris Cedex 16, France
[2] LSV, ENS de Cachan & CNRS,
61 av. du Président Wilson, F-94235 Cachan Cedex, France

Abstract. In the classical framework of formal languages, a refinement operation is modeled by a substitution and an abstraction by an inverse substitution. These mechanisms have been widely studied, because they describe a change in the specification level, from an abstract view to a more concrete one, or conversely. For timed systems, there is up to now no uniform notion of substitutions. In this paper, we study the timed substitutions in the general framework of signal-event languages, where both signals and events are taken into account. We prove that regular signal-event languages are closed under substitutions and inverse substitutions.

1 Introduction

Refinements and abstractions. Operations of refinements and abstractions are essential tools for the design and the study of systems, real-time or not. They allow to consider a given system at different levels of abstractions. For instance, some procedure or function can simply be viewed at some abstract level as a single action, and can be later expanded into all its possible behaviours at some more concrete level. Or conversely, a set of behaviours are merged together and replaced by a single action, in order to obtain a more abstract description.

These operations can be formally modeled by substitution and inverse substitution respectively. Therefore, substitutions have been extensively studied in the untimed framework, with the underlying idea that interesting classes of languages have to be closed under these operations.

Timed languages. In order to accept timed words, the model of timed automata was first proposed in [1, 2]. It has been widely studied for the last fifteen years and successfully applied to industrial cases. For this model, an observation, called a time-event word, may be viewed as an alternating sequence of waiting times and instantaneous actions. A timed automaton is a finite automaton extended with variables called clocks, designed to recognize time-event words: time elapses while the control stays in a given node and an event is observed when a discrete transition occurs.

Another model was introduced by [3], and further studied in [10, 4, 11] with the aim of describing hardware systems. In this case, an observation is a signal word, i.e., a sequence of factors a^d, where a is a signal and d is its duration. The original model of timed automata was then modified to fit this setting: a signal is emitted while the automaton stays in some state and no event is produced when a discrete transition is fired. In this framework, when a transition occurs between two states with the same signal a, we obtain a^{d_1} followed by a^{d_2}, which are merged into $a^{d_1+d_2}$. This phenomenon is called stuttering.

It was noticed in [4] that both approaches are complementary and can be combined in an algebraic formalism to obtain the so-called signal-event monoid. Timed automata can be easily adapted to take both signals and events into account, thus yielding signal-event automata: states emit signals and transitions produce events.

We consider in this paper both finite and infinite behaviors of signal-event automata and we also include unobservable events (ε-transitions) and hidden signals (τ-labeled states). It turns our that these features are very useful, for instance for handling abstractions. They also allow us to get as special cases the initial models of timed automata and signal automata.

Timed substitutions. Timed substitutions were studied in [8] for regular transfinite time-event languages. In [8], although no signal appear explicitely, actions are handled in a way similar (but not identical) to signals, *without* stuttering. Here we restrict the study of substitutions to finite and ω-sequences but we do handle signal stuttering which is a major difficulty.

Our contribution. The aim of this paper is to study the closure by substitutions and inverse substitutions of the families SEL_ε and SEL of languages accepted by signal-event automata, with or without ε-transitions. We prove that the class SEL_ε is closed under arbitrary substitutions and under arbitrary inverse substitutions. These closure properties are not verified by the class SEL in general. Nevertheless, we show that SEL is closed under inverse substitutions acting on events only, i.e., leaving signals unchanged, and we give a sufficient condition for its closure under substitutions. These results again show the robustness of the class SEL_ε, which is in favour of signal-event automata including ε-transitions.

Outline of the paper. We first give in Section 2 precise definitions of finite and infinite signal-event languages, with the corresponding notion of signal-event automata and we recall some technical results on signal-event automata that will be crucial for further proofs. In Section 3, we define timed substitutions which are duration preserving mappings. We then study in Section 4 the closures of the classes SEL and SEL_ε under reconizable substitutions and their inverses.

For lack of space, this paper does not contain the proofs of the correctness of the different automata constructions we proposed. These proofs are available in the technical report [6].

2 Signal-event words and signal-event automata

Let Z be any set. We write Z^* (respectively Z^ω) the set of finite (respectively infinite) sequences of elements in Z, with ε for the empty sequence, and $Z^\infty = Z^* \cup Z^\omega$ the set of all sequences of elements in Z. The set Z^∞ is equipped with the usual partial concatenation defined from $Z^* \times Z^\infty$ to Z^∞.

Throughout this paper, we consider a time domain $\mathbb{T}$ which can be either the set $\mathbb{N}$ of natural numbers, the set $\mathbb{Q}_+$ of non-negative rational numbers or the set $\mathbb{R}_+$ of non-negative real numbers and we set $\overline{\mathbb{T}} = \mathbb{T} \cup \{\infty\}$.

2.1 Signal-event words

We now describe the most general class of systems where both piecewise-constant signals and discrete events can occur, based on the signal-event monoid defined in [4]. We consider two finite alphabets Σ_e and Σ_s, with $\Sigma = \Sigma_e \cup (\Sigma_s \times \mathbb{T})$: an element in Σ_e is the label of an instantaneous event, while a pair $(a, d) \in \Sigma_s \times \mathbb{T}$, written a^d, associates a duration d with a signal a. Moreover, Σ_s includes the special symbol τ for an internal (or hidden) signal, the purpose of which is to represent a situation where no signal can be observed.

Intuitively, *signal-event words* (SE-words for short and sometimes called *timed words*) correspond to sequences obtained from Σ^∞ by merging consecutive identical signals and removing internal τ-signals with duration 0. But note that signals different from τ may have a null duration.

Formally, the partial monoid of signal-event words is the quotient $\Sigma^\infty / \approx$ where $\approx$ is the congruence (with respect to the partial concatenation on Σ^∞) generated by

$$\begin{cases} \tau^0 \approx \varepsilon & \text{and} \\ \prod_{i \in I} a^{d_i} \approx \prod_{j \in J} a^{d'_j} & \text{if } \sum_{i \in I} d_i = \sum_{j \in J} d'_j \end{cases}$$

where the index sets I and J above may be infinite. The partial monoid $\Sigma^\infty / \approx$ will be denoted $SE(\Sigma, \mathbb{T})$ or simply $SE(\Sigma)$ or SE when there is no ambiguity. We write a^∞ for the equivalence class of any sequence of the form $\prod_{i \geq 1} a^{d_i}$, where $\sum_{i \geq 1} d_i = \infty$. Note that for two words of the forms ua^d and $a^{d'}v$ with $d < \infty$, the concatenation is $ua^{d+d'}v$.

A finite or infinite sequence in $\Sigma^\infty \cup \Sigma^* \cdot (\Sigma_s \times \{\infty\})$ which does not contain τ^0 and such that two consecutive signals are distinct is said to be in *normal form* (NF). SE-words are often identified with sequences in normal form. A SE-word is *finite* if its normal form is a finite sequence (even if it ends with a^∞).

A duration can be associated with each element of Σ by: $\|a\| = 0$ if $a \in \Sigma_e$ and $\|a^d\| = d$ if $a \in \Sigma_s$ and $d \in \overline{\mathbb{T}}$, so that the duration of a sequence $w = s_1 s_2 \cdots$ in Σ^∞ is $\|w\| = \sum_{i \geq 1} \|s_i\| \in \overline{\mathbb{T}}$. Note that the duration restricted to finite SE-words with finite durations is a morphism from Σ^* into $(\mathbb{T}, +)$. A *Zeno* word is a SE-word with finite duration and whose normal form is infinite. A *signal-event language* (or *timed language*) is a set of SE-words.

Example 1. Let $\Sigma_e = \{f, g\}$ and $\Sigma_s = \{a, b\}$. The SE-word $w = a^3 f f g \tau^{4.5} a^1 b^5$ can be viewed as the following sequence of observations: first, the signal a during 3 time units, then a sequence of three instantaneous events ffg, then some unobservable signal during 4.5 time units, again the signal a during 1 time unit and then the signal b during 5 time units. The total duration of w is 13.5. For infinite SE-words, we have for instance: $a^3 g f a^1 \prod_{i \geq 1} a^2 \approx a^1 a^2 g f \prod_{i \geq 1} a^4$ and the normal form is written $a^3 g f a^\infty$. Note also that an infinite timed sequence in Σ^ω may be a finite SE-word with finite duration: $\prod_{i \geq 0} a^{1/2^i} \approx a^2$.

2.2 Signal-event (timed) automata

Our model of *signal-event automata* (also called *timed automata* in the sequel) is a variant of the basic models proposed in the literature, integrating both instantaneous and durational semantics: signals are associated with the control states, while instantaneous events occur when the system switches between two states.

Clocks and guards. Let X be a set of variables with values in $\mathbb{T}$, called clocks. The set $\mathcal{C}(X)$ of guards or clock constraints over X consists of conjunctions of atomic formulas $x \bowtie c$, for a clock x, a constant $c \in \mathbb{T}$ and a binary operator $\bowtie$ in $\{<, \leq, =, \geq, >\}$.

A clock valuation $v : X \to \mathbb{T}$ is a mapping that assigns to each clock x a time value $v(x)$. The set of all clock valuations is $\mathbb{T}^X$. We write $v \models g$ when the clock valuation v satisfies the clock constraint g. If t is an element of $\mathbb{T}$ and α a subset of X, the valuations $v + t$ and $v[\alpha]$ are defined respectively by $(v + t)(x) = v(x) + t$, for each clock x in X and $(v[\alpha])(x) = 0$ if $x \in \alpha$, and $(v[\alpha])(x) = v(x)$ otherwise.

Signal-event (timed) automata. A Büchi signal-event automaton over the time domain $\mathbb{T}$ is a tuple $\mathcal{A} = (\Sigma_e, \Sigma_s, X, Q, Q_0, F, R, I, \ell, \Delta)$, where Σ_e and Σ_s are alphabets of events and signals, X is a finite set of $\mathbb{T}$-valued clocks, Q is a finite set of control states, $Q_0 \subseteq Q$ is a subset of initial states, $F \subseteq Q$ is a subset of final states and $R \subseteq Q$ corresponds to a Büchi acceptance condition. The mapping $I : Q \to \mathcal{C}(X)$ associates with a state $q \in Q$ an *invariant* $I(q)$ being a conjunction of constraints of the form $x \bowtie c$, with $\bowtie \in \{<, \leq\}$, and $\ell : Q \to \Sigma_s$ associates a signal with each state.

The set of transitions is $\Delta \subseteq Q \times \mathcal{C}(X) \times \Sigma_e \cup \{\varepsilon\} \times \mathcal{P}(X) \times Q$. A transition, also written $q \xrightarrow{g,a,\alpha} q'$, is labeled by a guard g, an instantaneous event in Σ_e or the unobservable event ε, and the subset α of clocks to be reset. When $a = \varepsilon$, it is called an ε-transition or a silent transition.

First examples of signal-event automata are given in Figure 1 (where double-circled nodes correspond to Büchi repeated states). The semantics of *SE*-automata will be given below. But intuitively,

- A *SE*-word is accepted by $\mathcal{A}_1$ if it is of the form $a^{d_1} b^{d_2}$ with $d_1 \geq 1$.
- A *SE*-word is accepted by $\mathcal{A}_2$ if it is of the form $a^{d_1} c a^{d_2} c \ldots$ with $d_i < 1$ for any i.
- Since the concatenation merges consecutive identical signals ($a^{d_1} a^{d_2} = a^{d_1 + d_2}$), the language accepted by $\mathcal{A}_3$ consists of the signal a emitted for a duration $d \geq 1$.
- $\mathcal{A}_4$ accepts the signal a emitted for a duration $d \leq 1$ (note that a^1 is accepted by an infinite run with successive durations $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\ldots$ for instance).

Semantics. In order to define the semantics of *SE*-automata, we recall the notions of path and timed run through a path. A path in $\mathcal{A}$ is a finite or infinite sequence of consecutive transitions:

$$P = q_0 \xrightarrow{g_1, a_1, \alpha_1} q_1 \xrightarrow{g_2, a_2, \alpha_2} q_2 \ldots, \text{ where } (q_{i-1}, g_i, a_i, \alpha_i, q_i) \in \Delta, \ \forall i > 0$$

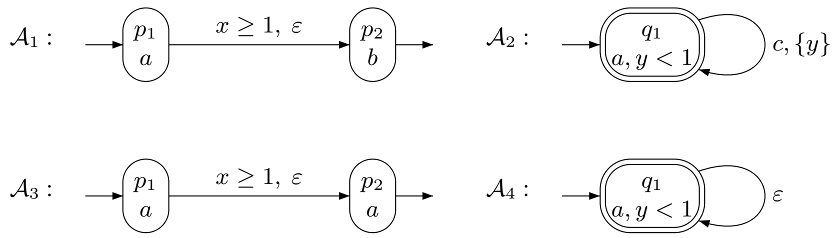


Fig. 1. Some signal automata.

The path is said to be *accepting* if it starts in an initial state ($q_0 \in Q_0$) and *either* it is finite and ends in a final state, *or* it is infinite and visits infinitely often a *repeated* state $q \in R$. A *run* of the automaton through the path P is a sequence of the form:

$$\langle q_0, v_0 \rangle \xrightarrow{d_0} \langle q_0, v_0 + d_0 \rangle \xrightarrow{a_1} \langle q_1, v_1 \rangle \xrightarrow{d_1} \langle q_1, v_1 + d_1 \rangle \xrightarrow{a_2} \langle q_2, v_2 \rangle \ldots$$

where

– $d_i \in \mathbb{T}$ for $i \geq 0$ and if P is finite with n transitions then the last step of the run must be $\langle q_n, v_n \rangle \xrightarrow{d_n} \langle q_n, v_n + d_n \rangle$, with $d_n \in \overline{\mathbb{T}}$,
– $(v_i)_{i \geq 0}$ are clock valuations such that $v_0(x) = 0$ for all $x \in X$, and for each $i \geq 0$, we have

$$\begin{cases} v_i + d \models I(q_i), & \forall d \in [0, d_i] \\ v_i + d_i \models g_{i+1} \\ v_{i+1} = (v_i + d_i)[\alpha_{i+1}] \end{cases}$$

Note that if d_i is finite, the condition about invariant $I(q_i)$ can be replaced simply by $v_i + d_i \models I(q_i)$.

The signal-event (timed) word generated by this run is simply (the equivalence class of) $\ell(q_0)^{d_0} a_1 \ell(q_1)^{d_1} a_2 \ell(q_2)^{d_2} \ldots$. The signal-event (timed) language accepted by $\mathcal{A}$ over the time domain $\mathbb{T}$ and the alphabet Σ, written $\mathcal{L}(\mathcal{A})$, is the set of SE-words generated by (finite or infinite) accepting runs of $\mathcal{A}$. Two automata $\mathcal{A}$ and $\mathcal{B}$ are *equivalent* if $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{B})$.

The set of all signal-event (timed) automata is denoted by SEA_ε whereas SEA is the set of all signal-event automata using transitions with observable events only (i.e. with labels in Σ_e instead of $\Sigma_e \cup \{\varepsilon\}$). The family of signal-event (timed) languages generated by some signal-event automaton in SEA_ε (respectively SEA) is denoted by SEL_ε (respectively SEL).

Remark 1. A *Zeno run* is an infinite run for which the time sequence defined by $t_i = \sum_{j \leq i} d_j$ for $i \geq 0$, is convergent (keeping the notations just above).
We did not include the non Zeno condition for runs, requiring that each infinite accepting run has an infinite duration. Thus, Zeno runs accepting finite words with finite

duration may occur. Although Zeno behaviours have been studied (see for instance [12, 9]), it has been argued that excluding Zeno runs directly from the semantics of timed automata is a realistic assumption, since they do not appear in "real" systems. However, the semantics of timed automata used by model-checking tools like UPPAAL do include Zeno runs while performing forward reachability analysis (this can be easily checked with an example). Hence, we think that the theory should include Zeno runs as well.

We state now two properties that will be used to prove the main results of this paper. The constructions related to the following two propositions can be found in [7]. It turns out that any SE-automaton can be transformed into an equivalent one, in which finite accepting runs correspond exactly to finite words with finite duration. Note that the transformation removes a particular type of Zeno runs, those which contain ultimately only ε-transitions and a single signal. But it keeps Zeno runs corresponding to infinite words of finite duration.

Proposition 1. *Let $\mathcal{A}$ be a SE-automaton. We can effectively construct an equivalent SE-automaton $\mathcal{A}'$ such that:*

(†) *no infinite run of $\mathcal{A}'$ accepts a finite word with finite duration, and*
(‡) *no finite run of $\mathcal{A}'$ accepts a word with infinite duration.*

The previous result is interesting in itself for implementation issues: when a finite word with finite duration is accepted by an infinite run, we build instead a finite accepting run for this word. Furthermore, conditions (†) and (‡) are crucial to prove the following closure property and are also used in the proof of Theorem 2.

Proposition 2. *The class SEL_ε is closed under intersection.*

If we consider signal-event automata where ε-transitions are not allowed, the treament of the intersection becomes much simpler. Indeed, the intersection of two SE-automata without ε-transitions can be done in a classical way, i.e. as a product of automata and a suitable treatment of the Büchi conditions. But in the general case, the construction of an automaton recognizing the intersection is much more tricky. It can be found in [7], together with references on related works.

3 Signal-event (timed) substitutions

Recall that substitutions are a suitable model for refinements. In the untimed framework, the image of each letter $a \in \Sigma$ is a given language L_a over another alphabet Σ' and a substitution is a morphism extending this mapping.

Dealing with timed words requires to preserve durations. Therefore, an instantaneous event must be replaced by SE-words with null duration, while a signal a with duration d must be replaced by SE-words of the same duration d. Formally, the new alphabet is also of the form $\Sigma' = \Sigma'_e \cup (\Sigma'_s \times \mathbb{T})$ and a substitution σ is defined by a family of SE-languages $(L_a)_{a \in \Sigma_e \cup \Sigma_s}$ such that:

- $L_a \subseteq (\Sigma'_e \cup (\Sigma'_s \times \{0\}))^*$ if $a \in \Sigma_e$,

– if $a \in \Sigma_s \setminus \{\tau\}$, then L_a is a SE-language of *non Zeno* SE-words over Σ'.

Using Zeno-words in substitutions may give rise to transfinite sequences, therefore we have excluded them from the languages L_a.

Moreover, we assume that the internal signal τ is never modified, so that we always have $L_\tau = \{\tau\} \times \overline{\mathbb{T}}$. Then for each $a \in \Sigma_e$, $\sigma(a) = L_a$ and for each $a^d \in \Sigma_s \times \overline{\mathbb{T}}$, $\sigma(a^d) = \{w \in L_a \mid \|w\| = d\}$. A substitution is thus a duration preserving mapping.

For a SE-word $v = v_1 v_2 \cdots$ in normal form over Σ, $\sigma(v)$ is the set of SE-words obtained from $\sigma(v_1)\sigma(v_2)\cdots$ by merging consecutive identical signals. Note that $w \in \sigma(v)$ can be written $w_1 w_2 \cdots$ with $w_i \in \sigma(v_i)$ for each $i \geq 1$. However this decomposition of w may not be in normal form. Finally, for a timed language L over Σ, we set $\sigma(L) = \cup_{v \in L} \sigma(v)$.

Example 2.

– Choosing $L_f = \{f\}$ for $f \in \Sigma_e$ and $L_a = \{\tau^0\} \cup \{a\} \times (\overline{\mathbb{T}} \setminus \{0\})$ for $a \in \Sigma_s$ leads to a substitution that cancel all signals with a null duration.
– Hiding some signal a is simulated by a substitution where $L_a = \{\tau\} \times \overline{\mathbb{T}}$.

Timed substitutions and morphisms. It should be noticed that, while substitutions are morphisms in the untimed framework, this is not the case with our definition. For instance, assume that a is a signal such that $L_a = \{b^2\}$, then $\sigma(a^1) = \emptyset$ and $\sigma(a^2) = \{b^2\} \neq \sigma(a^1)\sigma(a^1)$.

We call *m-substitution* a SE-substitution which is a morphism with respect to the partial concatenation. We have the following characterization:

Proposition 3. *Let σ be a SE-substitution, given by a family $(L_a)_{a \in \Sigma_e \cup \Sigma_s}$. Then, σ is a morphism if and only if for each signal $a \in \Sigma_s$ we have*

1. *L_a is closed under concatenation: for all $u, v \in L_a$ with $\|u\| < \infty$, we have $uv \in L_a$,*
2. *L_a is closed under decomposition: for each $v \in L_a$ with $\|v\| = d$, for all $d_1 \in \mathbb{T}$, $d_2 \in \overline{\mathbb{T}}$ such that $d = d_1 + d_2$, there exist $v_i \in L_a$ with $\|v_i\| = d_i$ such that $v = v_1 v_2$.*

This proposition is easy to prove and shows that rather restrictive conditions should be added on the languages L_a to obtain morphisms.

The notion of abstraction is fundamental in the study of systems, and in particular in their verification. It consists in replacing a set of behaviors with a single action in order to obtain a smaller system, simpler to study and to understand. The abstraction operation is thus the inverse of refinement. As in the untimed case, inverse substitutions provide a suitable model for abstractions in our framework.

For a substitution σ, the inverse substitution σ^{-1} is the operation defined for a language $L' \subseteq SE(\Sigma')$ by $\sigma^{-1}(L') = \{v \in SE(\Sigma) \mid \sigma(v) \cap L' \neq \emptyset\}$.

4 Recognizable substitutions

We now focus on recognizable substitutions, whose associated languages are defined by SE-automata. Formally, a substitution σ defined by a family of languages $(L_a)_{a\in\Sigma_e\cup\Sigma_s}$ is a SEL_ε-substitution (a SEL-substitution resp.) if $L_a \in SEL_\varepsilon$ (resp. $L_a \in SEL$) for each $a \in \Sigma_e \cup \Sigma_s$.

The aim of this paper is precisely to investigate the closure of the two classes of SE-languages SEL and SEL_ε under recognizable substitutions and their inverse.

We first consider the special cases of renaming and event-hiding. Formally, a renaming is simply a substitution such that, for each $f \in \Sigma_e$, $L_f = \{g\}$ for some $g \in \Sigma'_e$, and for each $a \in \Sigma_s \setminus \{\tau\}$, $L_a = \{b\} \times \overline{\mathbb{T}}$ for some $b \in \Sigma'_s$. An event-hiding is a substitution such that $L_f = \{f\}$ or $L_f = \{\varepsilon\}$ for $f \in \Sigma_e$ and $L_a = \{a\} \times \overline{\mathbb{T}}$ for $a \in \Sigma_s$. We have:

Proposition 4.

1. *The classes SEL and SEL_ε are closed under renaming,*
2. *The class SEL_ε is closed under event-hiding whereas SEL is not,*

Proof. Point 1 is straightforward. For the second point, the closure of SEL_ε under event-hiding was already noticed in [1] and the result easily extends to our framework: in order to hide the event a, one replaces a-labelled transitions by ε-transitions.

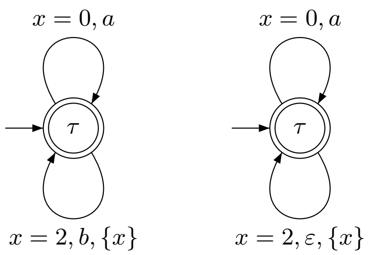
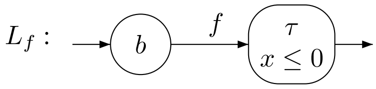
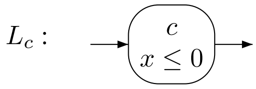


Fig. 2. Timed automata for L_1 and $h(L_1)$.

For the class SEL, recall from [5] that the language $Even = ((\tau^2)^*a)^\infty$ cannot be accepted by a timed automaton without ε-transitions. Consider the language $L_1 = ((\tau^2 b)^* a)^\omega \cup ((\tau^2 b)^* a)^* (\tau^2 b)^\omega$ accepted by the automaton on the left of Figure 2, with no final state, where the unique state is labeled τ and is both initial and repeated. Hiding the b's in L_1 yields the language $h(L_1)$ accepted by the automaton in SEA_ε on the right of Figure 2. Since $h(L_1)$ is the analogous of $Even$ in our framework, it is not in the class SEL. $\square$

The class SEL is not closed under SEL-substitution. Indeed, take $L_f = \{b^0 f\}$ and $L_c = \{c^0\}$ which are accepted by the automata below and are therefore both in SEL. Consider also $L = \{c^0 f\} \in SEL$. Then $\sigma(L) = \{c^0 b^0 f\}$ is not in SEL since it contains a SE-word with two consecutive distinct signals. The next result gives a sufficient condition on a substitution for the closure property of SEL to hold.





Theorem 1. *Let L be a language in SEL and σ a SEL-substitution such that for each $f \in \Sigma_e$ the language L_f contains only SE-words starting and ending with (instantaneous) events from Σ'_e. Then $\sigma(L)$ belongs to SEL.*

Proof. We prove the theorem for a substitution acting only on events or acting only on signals. The general case is obtained using compositions of such elementary substitutions and of renaming.

We first show how to handle substitution of events. So let σ be a SEL-substitution satisfying the condition of the theorem and which is defined by a family $(L_a)_{a\in\Sigma_e\cup\Sigma_s}$ of SEL-languages. Since σ acts only on events, we have $L_a = \{a\} \times \overline{\mathbb{T}}$ for all $a \in \Sigma_s$. For $f \in \Sigma_e$, the language L_f is accepted by some SEA-automaton $\mathcal{A}_f$ with only one clock x_f which is never tested or reset, with no repeated states, where all the guards are `true` and where all final states carry the invariant $x_f \leq 0$ in order to ensure that each accepted SE-word has a null duration. Moreover, from the hypothesis on σ, we can assume that all initial and final states are labelled τ.

Now, let $L \subseteq SE(\Sigma)$ be accepted by a SEA-automaton $\mathcal{A}$. We build from $\mathcal{A}$ a SEA-automaton $\mathcal{A}'$ accepting $\sigma(L)$ as follows. For each state q, we consider a new copy $\mathcal{A}_f^q$ of $\mathcal{A}_f$ and we replace any transition (p, g, f, α, q) by the following set of transitions:

- for each transition $(q_0, \mathtt{true}, b, \emptyset, q_1)$ in $\mathcal{A}_f^q$ with q_0 initial, we add the transition $(p, g, b, \alpha \cup \{x_f\}, q_1)$ to $\mathcal{A}'$,
- for each transition $(q_1, \mathtt{true}, b, \emptyset, q_2)$ in $\mathcal{A}_f^q$ with q_2 final, we add the transition $(q_1, x_f = 0, b, \emptyset, q)$ to $\mathcal{A}'$,
- for each transition $(q_0, \mathtt{true}, b, \emptyset, q_2)$ in $\mathcal{A}_f^q$ with q_0 initial and q_2 final, we add the transition (p, g, b, α, q) to $\mathcal{A}'$,
- all states and transitions of $\mathcal{A}_f^q$ are kept unchanged in $\mathcal{A}'$.

Again, the clock operations on x_f ensure an instantaneous traversal of $\mathcal{A}_f^q$. The initial, final and repeated states remain those of $\mathcal{A}$. Clearly $\mathcal{A}'$ is a SEA-automaton and we can show that it accepts $\sigma(L)$.

We now handle substitution of signals. Let σ be a SEL-substitution defined by a family $(L_a)_{a\in\Sigma_e\cup\Sigma_s}$ of SEL-languages. We assume that σ acts only on signals, i.e., $L_f = \{f\}$ for each $f \in \Sigma_e$. For $a \in \Sigma_s$, the language L_a is accepted by some SEA-automaton $\mathcal{A}_a = (\Sigma'_e, \Sigma'_s, X_a, Q_a, Q_a^0, F_a, R_a, I_a, \ell_a, \Delta_a)$. We assume that all the X_a's are pairwise disjoints.

Let $\mathcal{A} = (\Sigma_e, \Sigma_s, X, Q, Q_0, F, R, I, \ell, \Delta)$ be a SEA-automaton accepting a language $L \subseteq SE(\Sigma)$. We assume that X is disjoint from the X_a's. We build from $\mathcal{A}$ a SEA-automaton $\mathcal{A}'$ accepting $\sigma(L)$ as follows. For each state p of $\mathcal{A}$ with label a, we consider a new copy $\mathcal{A}_p$ of $\mathcal{A}_a$ in which the invariant $I(p)$ is added to all states: $I_p(r_p) = I_a(r) \wedge I(p)$ if r_p is the copy of state $r \in Q_a$. To build $\mathcal{A}'$, we start with the disjoint union of all the $\mathcal{A}_p$'s and we add *switching* transitions:

> If (p, g, f, α, q) is a transition of $\mathcal{A}$ then for each final state r_p of $\mathcal{A}_p$ and each initial state s_q of $\mathcal{A}_q$ we add the transition $(r_p, g, f, \alpha \cup X_{\ell(q)}, s_q)$ to $\mathcal{A}'$.

The initial states of $\mathcal{A}'$ are the initial states of all $\mathcal{A}_p$ such that $p \in Q_0$. The final states of $\mathcal{A}'$ are the final states of all $\mathcal{A}_p$ such that $p \in F$. An infinite path of $\mathcal{A}'$ is accepting if

– either it uses infinitely many switching transitions $(r_p, g, f, \alpha \cup X_{\ell(q)}, s_q)$ with $p \in R$,
– or else, it stays ultimately in some $\mathcal{A}_p$ with $p \in F$ and it visits R_p infinitely often.

We can easily transform the automaton $\mathcal{A}'$ in order to get a classical Büchi condition if needed. We can check that $\mathcal{A}'$ accepts $\sigma(L)$. $\square$

We will now show that the class SEL_ε is closed under SEL_ε-substitutions. The construction for the substitution of signals given in the previous proof does not work. Indeed, by definition, a substitution must be applied to a word in normal form. The difficulty comes from the fact that in the automaton for L, a factor a^d of some normal form may be generated by a path with several a-labelled states and even τ-labelled states that are crossed instantaneously if all these states are linked by ε-transitions. So we cannot simply replace each a-labelled state by a copy of $\mathcal{A}_a$.

To circumvent this difficulty, we use a proof technique inspired from rational transductions and that can be applied to establish the closure of the class SEL_ε both under SEL_ε-substitutions and their inverse. Hence, we state and prove both results simultaneously. It should be noted that these closure properties hold for arbitrary substitutions, showing once again the robustness of the class SEL_ε.

Theorem 2. *The class SEL_ε is closed under SEL_ε-substitution and inverse SEL_ε-substitution.*

Proof. Let σ be a SEL_ε-substitution from $SE(\Sigma)$ to $SE(\Sigma')$ given by a family of automata $(\mathcal{A}_a)_{a \in \Sigma_e \cup \Sigma_s}$, with $\mathcal{A}_a = (\Sigma'_e, \Sigma'_s, X_a, Q_a, Q_a^0, F_a, R_a, I_a, \ell_a, \Delta_a) \in SEA_\varepsilon$. We assume that these automata satisfy condition (†) of Proposition 1.

We use the same technique to prove both closure properties. We show that $\sigma(L)$ and $\sigma^{-1}(L)$ can both be expressed as a projection of the intersection of a SEL_ε-language with an inverse projection of L. This is in the spirit of rational transductions for classical word languages.

Consider a new alphabet $\hat{\Sigma} = \hat{\Sigma}_e \cup \hat{\Sigma}_s \times \mathbb{T}$ with $\hat{\Sigma}_e = \Sigma_e \uplus \Sigma'_e$ ($\uplus$ is the disjoint union) and $\hat{\Sigma}_s = \Sigma_s \times \Sigma'_s$ (we identify (τ, τ) with the unobservable signal τ). The projections π_1 and π_2 are the morphisms defined by:

$\pi_1(f) = f$ and $\pi_2(f) = \varepsilon$ if $f \in \Sigma_e$,
$\pi_1(f) = \varepsilon$ and $\pi_2(f) = f$ if $f \in \Sigma'_e$,
$\pi_1((a,b)^d) = a^d$ and $\pi_2((a,b)^d) = b^d$ if $(a,b)^d \in \Sigma_s \times \Sigma'_s \times \overline{\mathbb{T}}$.

With this definition we have $\pi_i((a,b)^{d_1+d_2}) = \pi_i((a,b)^{d_1})\pi_i((a,b)^{d_2})$. Note that projection π_i is a composition of an event-hiding and a signal-renaming. By Proposition 4 we deduce that the projection by π_i of a SEL_ε language is again in the class SEL_ε.

For $L \subseteq SE(\Sigma)$, we let $\pi_1^{-1}(L) = \{w \in SE(\hat{\Sigma}) \mid \pi_1(w) \in L\}$. We define similarly $\pi_2^{-1}(L)$ for $L \subseteq SE(\Sigma')$. We will show later that if L is recognizable then so is $\pi_i^{-1}(L)$. We will also define a recognizable language $M \subseteq SE(\hat{\Sigma})$ with the following properties:

1. for each $w \in M$, we have $\pi_2(w) \in \sigma(\pi_1(w))$,
2. for each $u \in SE(\Sigma)$ and $v \in \sigma(u)$, there exists $w \in M$ such that $u = \pi_1(w)$ and $v = \pi_2(w)$.

Then, for $L \subseteq SE(\Sigma)$, we have $\sigma(L) = \pi_2(\pi_1^{-1}(L) \cap M)$. Indeed, Let $v \in \sigma(L)$ and let $u \in L$ with $v \in \sigma(u)$. Using property 2 of M we find $w \in M$ with $\pi_1(w) = u$ and $\pi_2(w) = v$. Then, $w \in \pi_1^{-1}(L) \cap M$ and $v \in \pi_2(\pi_1^{-1}(L) \cap M)$. Conversely, let $v \in \pi_2(\pi_1^{-1}(L) \cap M)$ and let $w \in \pi_1^{-1}(L) \cap M$ with $\pi_2(w) = v$. By definition, we have $u = \pi_1(w) \in L$ and using property 1 of M, we get $v \in \sigma(u) \subseteq \sigma(L)$.

Similarly, for $L \subseteq SE(\Sigma')$, we have $\sigma^{-1}(L) = \pi_1(\pi_2^{-1}(L) \cap M)$. Indeed, Let $u \in \sigma^{-1}(L)$ and let $v \in \sigma(u) \cap L$. Using property 2 of M we find $w \in M$ with $\pi_1(w) = u$ and $\pi_2(w) = v$. Then, $w \in \pi_2^{-1}(L) \cap M$ and $u \in \pi_1(\pi_2^{-1}(L) \cap M)$. Conversely, let $u \in \pi_1(\pi_2^{-1}(L) \cap M)$ and let $w \in \pi_2^{-1}(L) \cap M$ with $\pi_1(w) = u$. By definition, we have $v = \pi_2(w) \in L$ and using property 1 of M, we get $v \in \sigma(u)$. Hence, $\sigma(u) \cap L \neq \emptyset$ and $u \in \sigma^{-1}(L)$.

We already know that SEL_ε-languages are closed under intersection (Proposition 2) and projections π_i. To conclude the proof of Theorem 2, it remains to show that they are also closed under inverse projections π_i^{-1} and to define the SEL_ε-language M with the properties above.

We show first that SEL_ε-langagues are closed under inverse projections π_i^{-1}. Let L be recognized by some automaton $\mathcal{A} = (\Sigma_e, \Sigma_s, X, Q, Q^0, F, R, I, \ell, \Delta) \in SEA_\varepsilon$. We build an automaton $\hat{\mathcal{A}}$ accepting $\pi_1^{-1}(L)$. The set of states is $\hat{Q} = Q \uplus Q \times \Sigma'_s$, with $\hat{Q}^0 = Q^0$ for initial states and $\hat{F} = F$ for final states. The set of clocks is $X \uplus \{z\}$. The labels and invariants are defined by $\hat{\ell}(q) = \tau$ and $\hat{I}(q) = I(q) \wedge (z \leq 0)$ for $q \in Q$, and $\hat{\ell}((q, b)) = (\ell(q), b)$ and $\hat{I}((q, b)) = I(q)$ for $(q, b) \in Q \times \Sigma'_s$. The set of transitions $\hat{\Delta}$ is defined by:

1. All transitions $(p, g, f, \alpha, q) \in \Delta$ are kept in $\hat{\Delta}$.
2. For all $f' \in \Sigma'_e$ and $q \in Q$, we put $(q, \mathtt{true}, f', \emptyset, q)$ in $\hat{\Delta}$.
3. For all $(q, b) \in Q \times \Sigma'_s$, we put $(q, \mathtt{true}, \varepsilon, \emptyset, (q, b))$, $((q, b), \mathtt{true}, \varepsilon, \{z\}, q)$ in $\hat{\Delta}$.

We use a generalized acceptance condition for infinite paths. By transforming the automaton we can get a classical Büchi condition if needed. An infinite path is accepting if it uses

- either infinitely many transitions (p, g, f, α, q) of type 1 with $q \in R$,
- or ultimately transitions of type 2 and 3 only around some state $q \in F$.

We show that $\mathcal{L}(\hat{\mathcal{A}}) = \pi_1^{-1}(L)$. Let w be a word over $\hat{\Sigma}$ accepted by $\hat{\mathcal{A}}$. We consider a run of $\hat{\mathcal{A}}$ for w through an accepting path $\hat{P}$. Erasing from this path all transitions of type 2 and 3 above, we obtain a path P of $\mathcal{A}$. If $\hat{P}$ is finite then it ends in a final state $q \in \hat{F} = F$ and P is also accepting since it ends in state q. If $\hat{P}$ uses infinitely many transitions (p, g, f, α, q) of type 1 with $q \in R$, then P is infinite and visits R infinitely often, hence it is also accepting. Finally, if $\hat{P}$ is infinite and uses ultimately transitions of type 2 and 3 only around some state $q \in F$, then P is finite and accepting since it ends in state q. In all cases, P is an accepting path of $\mathcal{A}$. We can show that $\pi_1(w)$ admits a run through P. Therefore, $\pi_1(w) \in L$.

Conversely, let $w \in SE(\hat{\Sigma})$ and assume that $\pi_1(w) \in L$ is accepted by a run $\langle q_0, v_0 \rangle \xrightarrow{d_0} \langle q_0, v_0 + d_0 \rangle \xrightarrow{f_1} \langle q_1, v_1 \rangle \xrightarrow{d_1} \langle q_1, v_1 + d_1 \rangle \xrightarrow{f_2} \langle q_2, v_2 \rangle \cdots$ through an accepting path $P = q_0 \xrightarrow{g_1, f_1, \alpha_1} q_1 \xrightarrow{g_2, f_2, \alpha_2} q_2 \cdots$ of $\mathcal{A}$. Then, we have $\pi_1(w) \approx a_0^{d_0} f_1 a_1^{d_1} f_2 \cdots$ with $a_i = \ell(q_i)$ for $i \geq 0$. We deduce that $w \approx w_0 f_1 w_1 f_1 w_2 \cdots$ with $\pi_1(w_i) = a_i^{d_i}$. Now, if w_i is finite with finite duration then we find a path $\hat{P}_i$ following the normal form of w_i, starting and ending in q_i, and using transitions of type 2 and 3 only. If w_i is infinite or with infinite duration then the path P must be finite ending in state $q_i \in F$ since P is accepting. Then, we find an *infinite* path $\hat{P}_i$ for w_i using only transitions of type 2 and 3 around q_i. Note that if w_i is finite with infinite duration, then it ends with b^∞ for some $b \in \Sigma'_s$ and we still need an infinite path ultimately alternating between states q_i and (q_i, b). Finally, $\hat{P} = \hat{P}_0 \xrightarrow{g_1, f_1, \alpha_1} \hat{P}_1 \xrightarrow{g_2, f_2, \alpha_2} \hat{P}_2 \cdots$ is a path in $\hat{\mathcal{A}}$ and it is easy to see that $\hat{P}$ is accepting. Moreover, we can show that w admits a run through $\hat{P}$ and therefore, w is accepted by $\hat{\mathcal{A}}$.

We turn now to the definition of M. For $f \in \Sigma_e$ and $a \in \Sigma_s \setminus \{\tau\}$, we define

$$\begin{aligned} M_f = \{w \in SE(\hat{\Sigma}) \mid w = (\tau, b_0)^0 f_1 (\tau, b_1)^0 f_2 \cdots (\tau, b_n)^0 \\ \text{with } b_0^0 f_1 b_1^0 f_2 \cdots b_n^0 \in \sigma(f)\} \cdot f \\ M_a = \{w \in SE(\hat{\Sigma}) \mid w = (a, b_0)^{d_0} f_1 (a, b_1)^{d_1} f_2 \cdots \\ \text{with } b_0^{d_0} f_1 b_1^{d_1} f_2 \cdots \in \sigma(a^{d_0 + d_1 + \cdots})\} \end{aligned}$$

We also let $M_\tau = \{(\tau, \tau)^d \mid d \in \overline{\mathbb{T}} \setminus \{0\}\}$. Note that for $w \in M_a$ with $a \in \Sigma_e \cup \Sigma_s$ we have $\pi_2(w) \in \sigma(\pi_1(w))$ as required by property 1 of M. Moreover, if $f \in \Sigma_e$ and $v \in \sigma(f)$ then there exists $w \in M_f$ such that $\pi_1(w) = f$ and $\pi_2(w) = v$. Similarly, if $a^d \in \Sigma_s \times \overline{\mathbb{T}}$ (with $d > 0$ if $a = \tau$) and $v \in \sigma(a^d)$ then there exists $w \in M_a$ such that $\pi_1(w) = a^d$ and $\pi_2(w) = v$.

Intuitively, M consists of finite or infinite products of words in $\bigcup_{a \in \Sigma_e \cup \Sigma_s} M_a$ except that, in order to ensure that the first projection is in normal form, we should not allow consecutive factors associated with the same signal. Formally, we define

$$M = \{w_1 w_2 \cdots \mid \exists a_1, a_2, \ldots \in \Sigma_e \cup \Sigma_s \text{ with } w_i \in M_{a_i} \text{ and } a_i \in \Sigma_s \Rightarrow a_{i+1} \neq a_i\}.$$

We show that M satisfies property 1. Let $w = w_1 w_2 \cdots \in M$ and let $a_1, a_2, \ldots \in \Sigma_e \cup \Sigma_s$ be such that $w_i \in M_{a_i}$ and $a_i \in \Sigma_s \Rightarrow a_{i+1} \neq a_i$. Then, $\pi_1(w) = \pi_1(w_1)\pi_1(w_2) \cdots$ is in normal form and we have seen above that $\pi_2(w_i) \in \sigma(\pi_1(w_i))$. Therefore, $\pi_2(w) \in \sigma(\pi_1(w))$ and property 1 is proved.

We show that M satisfies property 2. Let $u \in SE(\Sigma)$ and $v \in \sigma(u)$. Write $u = u_1 u_2 \cdots$ in normal form and $v = v_1 v_2 \cdots$ with $v_i \in \sigma(u_i)$. Let $a_i = u_i$ if $u_i \in \Sigma_e$ and $a_i = a$ if $u_i = a^d \in \Sigma_s \times \overline{\mathbb{T}}$. Since the product $u = u_1 u_2 \cdots$ is in normal form, we have $a_i \in \Sigma_s \Rightarrow a_{i+1} \neq a_i$ and $u_i = \tau^d \Rightarrow d > 0$. Since $v_i \in \sigma(u_i)$, we have seen above that there exists $w_i \in M_{a_i}$ such that $\pi_1(w_i) = u_i$ and $\pi_2(w_i) = v_i$. Then, $w = w_1 w_2 \cdots \in M$ and $\pi_1(w) = u$ and $\pi_2(w) = v$ as required by property 2.

It remains to show that M is recognizable by some SEA_ε-automaton. For each $a \in \Sigma_e \cup \Sigma_s$ we first show that M_a is accepted by some automaton $\hat{\mathcal{A}}_a \in SEA_\varepsilon$ derived from $\mathcal{A}_a$. This is clear for $a = \tau$. Note that $\hat{\mathcal{A}}_\tau$ needs two states to ensure the positive duration required by M_τ. For $a \in \Sigma_s \setminus \{\tau\}$, the automaton $\hat{\mathcal{A}}_a$ is simply a copy of $\mathcal{A}_a$, with new label $(a, \ell_a(q))$ for $q \in Q_a$. For $f \in \Sigma_e$, the set of states is $\hat{Q}_f = Q_f \uplus \{q_f\}$, where q_f is a new state, which is also the only final state. The label of $q \in Q_f$ is $(\tau, \ell_f(q))$, the label of q_f is (τ, τ) and its invariant is $x_f \leq 0$ where x_f is the clock ensuring instantaneous traversal of $\mathcal{A}_f$. The transitions are those in $\mathcal{A}_f$, to which we add (q, f, q_f) for any state q which was final in $\mathcal{A}_f$. Note that, since $\mathcal{A}_a$ satisfies (†) for $a \in \Sigma_e \cup \Sigma_s$, then so does $\hat{\mathcal{A}}_a$.

Since M is essentially the iteration of the languages M_a, it should be clear that $M \in SEL_\varepsilon$. The SEA_ε-automaton $\mathcal{B}$ recognizing M is the disjoint union of the automata $\hat{\mathcal{A}}_a$ to which we add ε-transitions allowing to switch between automata: if p is a final state of $\hat{\mathcal{A}}_a$ and q is an initial state of $\hat{\mathcal{A}}_b$ and $a \in \Sigma_s \Rightarrow b \neq a$ then we add the transition $(p, \mathtt{true}, \varepsilon, X_b, q)$. All initial (resp. final, repeated) states of the $\hat{\mathcal{A}}_a$'s are initial (resp. final, repeated) in $\mathcal{B}$. But we also need to accept runs that switch infinitely often between the $\hat{\mathcal{A}}_a$'s, i.e., taking infinitely many switching ε-transitions. If needed, it is easy to transform the automaton so that it uses classical Büchi condition to accept also these runs.

We can show that any SE-word accepted by $\mathcal{B}$ is in M. Condition (†) is needed to prove the converse. Indeed, a finite SE-word $v \in M_a$ with $\|v\| < \infty$ could appear as an internal factor of some SE-word $uvw \in M$ with $w \neq \varepsilon$. If v could only be accepted by an infinite run in $\hat{\mathcal{A}}_a$, then we would not be able to build a (non transfinite) run for uvw in $\mathcal{B}$. ☐

The class SEL is not closed under inverse SEL-substitutions. Indeed, assume $\Sigma_s = \Sigma'_s = \{a, b\}$, $\Sigma_e = \Sigma'_e = \{f\}$ and let σ be the SEL-substitution defined by $L_a = \{a^1 f\}$, $L_b = \{b^0\}$ and $L_f = \{f\}$. Then the inverse image by σ of the SEL-language $\{a^1 f b^0\}$ is the language $\{a^1 b^0\}$ which is not in the class SEL. Nevertheless sufficient conditions on the substitutions can be proposed to ensure the closure of the class SEL under inverse substitutions.

Theorem 3. *The class SEL is closed under inverse SEL-substitutions acting on events only.*

Proof. Let σ be a substitution defined by a family $(L_a)_{a \in \Sigma_e \cup \Sigma_s}$ of SEL-languages. We assume that σ acts only on events, i.e., $L_a = \{a\} \times \mathbb{T}$ for all $a \in \Sigma_s$. Let $\mathcal{A}_f = (\Sigma'_e, \Sigma'_s, \{x_f\}, Q_f, Q^0_f, F_f, \emptyset, I_f, \ell_f, \Delta_f)$ for $f \in \Sigma_e$ be an automaton without ε-transitions accepting L_f. Since the guards and resets on the transitions of $\mathcal{A}_f$ are always $\mathtt{true}$ and $\emptyset$ respectively, we write a transition of $\mathcal{A}_f$ simply (r, f', s) to simplify the notation. We consider now a language $L \in SEL$, recognized by some automaton without ε-transition $\mathcal{A}_2 = (\Sigma'_e, \Sigma_s, X, Q, Q^0, F, R, I, \ell, \Delta) \in SEA$.

We build an automaton $\mathcal{A}_1$ in SEA accepting $\sigma^{-1}(L)$ essentially by changing the transitions of $\mathcal{A}_2$. If there is a path P in $\mathcal{A}_2$ from p to q having some instantaneous run for some word in $\sigma(f)$ then we add to $\mathcal{A}_1$ a transition (p, g, f, α, q) with a suitable guard g and reset α. The difficulty is to compute a suitable pair (g, α) for each triple (p, q, f).

Given a guard g and a subset of clocks α, we define the restriction of g by α, written $g[\alpha]$, as the guard g where all clocks from α have been replaced by 0. For instance, if g is $(x < 3) \wedge (y > 2)$ and $\alpha = \{x, z\}$ then $g[\alpha]$ is (equivalent to) $(y > 2)$. We let $\mathcal{G}$ be the smallest set of guards including all guards of $\mathcal{A}_2$ and closed under conjunctions and restrictions. Formally $\mathcal{G}$ is not a finite set, but it can be identified with its finite quotient under equivalence of formulae: two formulae φ and ψ are equivalent if $v \models \varphi$ iff $v \models \psi$ for all valuations v. The set $\Gamma = \mathcal{G} \times \mathcal{P}(X)$ is thus a finite monoid, with $(True, \emptyset)$ as neutral element, for the associative composition:

$$(g_1, \alpha_1) \cdot (g_2, \alpha_2) = (g_1 \wedge g_2[\alpha_1], \alpha_1 \cup \alpha_2).$$

Finally, we define a morphism $\gamma : \Delta^* \mapsto \Gamma$ by $\gamma((p, g, f, \alpha, q)) = (g, \alpha)$.

Let $P = q_0 \xrightarrow{g_1, b_1, \alpha_1} q_1 \xrightarrow{g_2, b_2, \alpha_2} \cdots \xrightarrow{g_n, b_n, \alpha_n} q_n$ be a path in $\mathcal{A}_2$. We define $\mathcal{W}(P) = \ell(q_0)^0 b_1 \ell(q_1)^0 b_2 \cdots b_n \ell(q_n)^0$. Now, given a triple (p, q, f) where $p, q \in Q$ and $f \in \Sigma_e$, we denote by $L^f_{p,q}$ the set of paths P from p to q in $\mathcal{A}_2$ with $\mathcal{W}(P) \in \sigma(f)$. We can build an automaton $\mathcal{B}^f_{p,q}$ recognizing the language $L^f_{p,q} \subseteq \Delta^*$. This is not difficult since we are dealing with automata without ε-transitions, hence we can perform a simple synchronized product as follows. The set of states is $Q' = \{(r, s) \in Q_f \times Q \mid \ell_f(r) = \ell(s)\}$, the set of initial states is $Q' \cap Q^0_f \times \{p\}$, the set of final states is $Q' \cap F_f \times \{q\}$. The transitions of $\mathcal{B}^f_{p,q}$ are the triples $((r_1, s_1), (s_1, g, b, \alpha, s_2), (r_2, s_2))$ with $(r_1, b, r_2) \in \Delta_f$ and $(s_1, g, b, \alpha, s_2) \in \Delta$.

We let $\Delta_R \subseteq \Delta^*$ be the set of sequences containing at least a transition ending in some repeated state of R. From the automaton $\mathcal{B}^f_{p,q}$, we can effectively compute a rational expression for the language $L^f_{p,q} \cap \Delta_R$. We deduce that we can effectively compute the finite set $\gamma(L^f_{p,q} \cap \Delta_R)$. Similarly, we can effectively compute the finite set $\gamma(L^f_{p,q} \setminus \Delta_R)$. These two sets are used to define the transitions of a new automaton $\mathcal{A}_1 = (\Sigma_e, \Sigma_s, X, Q \uplus \overline{Q}, Q^0, F, \overline{Q}, I, \ell, \Delta_1)$ in SEA:

$$\begin{aligned} \Delta_1 = \quad & \{(p, g, f, \alpha, q), (\overline{p}, g, f, \alpha, q) \mid (g, \alpha) \in \gamma(L^f_{p,q} \setminus \Delta_R)\} \\ & \cup \{(p, g, f, \alpha, \overline{q}), (\overline{p}, g, f, \alpha, \overline{q}) \mid (g, \alpha) \in \gamma(L^f_{p,q} \cap \Delta_R)\}. \end{aligned}$$

The automaton $\mathcal{A}_1$ can therefore be effectively computed from the automata $\mathcal{A}_2$ and $(\mathcal{A}_f)_{f \in \Sigma_e}$. We can show that $L(\mathcal{A}_1) = \sigma^{-1}(L)$. Therefore, the language $\sigma^{-1}(L) \in SEL$ and Theorem 3 is proved. $\square$

5 Conclusion

We have shown in this paper that the class SEL of signal-event languages is not closed under arbitrary SEL-substitutions and inverse SEL-substitutions but that natural sufficient conditions ensure closure properties for this class.

But our main contribution is to propose effective constructions to prove the closure of the larger class SEL_ε under arbitrary SEL_ε-substitutions and inverse SEL_ε-substitutions. We give these constructions in the general framework of signal-event automata and languages. The usual cases of event languages [1, 2] or signal languages [3, 10, 4, 11] are particular cases for which the interested reader will check that simplified constructions can be given.

References

1. R. Alur and D.L. Dill. Automata for modeling real-time systems. In *Proceedings of ICALP'90*, number 443 in LNCS, pages 322–335. Springer, 1990.
2. R. Alur and D.L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.
3. E. Asarin, P. Caspi, and O. Maler. A Kleene theorem for timed automata. In *Proceedings of LICS'97*, pages 160–171. IEEE Comp. Soc. Press, 1997.
4. E. Asarin, P. Caspi, and O. Maler. Timed regular expressions. *Journal of the ACM*, 49(2):172–206, 2002.
5. B. Bérard, V. Diekert, P. Gastin, and A. Petit. Characterization of the expressive power of silent transitions in timed automata. *Fundamenta Informaticae*, 36:145–182, 1998.
6. B. Bérard, P. Gastin, and A. Petit. Timed substitutions for regular signal-event languages. Research Report LSV-06-04, Laboratoire Spécification et Vérification, ENS Cachan, France, February 2006.
7. B. Bérard, P. Gastin, and A. Petit. Intersection of regular signal-event (timed) languages. In *Proceedings of FORMATS'06*, number ??? in LNCS, pages ??–??. Springer, 2006.
8. B. Bérard and C. Picaronny. Accepting Zeno words: a way toward timed refinements. *Acta Informatica*, 37(1):45–81, 2000.
9. P.J.L. Cuijpers, M.A. Reniers, and A.G. Engels. Beyond zeno-behaviour. Technical Report CSR 01-04, Department of Computing Science, University of Technology, Eindhoven, 2001.
10. C. Dima. Real-Time Automata and the Kleene Algebra of Sets of Real Numbers. In *Proceedings of STACS'2000*, number 1770 in LNCS, pages 279–289. Springer, 2000.
11. J. Durand-Lose. A Kleene theorem for splitable signals. *Information Processing Letters*, 89:237–245, 2004.
12. M.R. Hansen, P.K. Pandya, and C. Zhou. Finite divergence. *Theoretical Computer Science*, 138:113–139, 1995.