Rational and Recognizable Complex Trace Languages*

Volker Diekert
Universität Stuttgart
Institut für Informatik
Breitwiesenstr. 20-22
D-7000 Stuttgart 80

Paul Gastin
Université Paris 6
LITP, Institut Blaise Pascal
4, place Jussieu
F-75252 Paris Cedex 05

Antoine Petit
Université Paris Sud
LRI, URA CNRS 410
Bât. 490
F-91405 Orsay Cedex

March 16, 1995

Abstract

Mazurkiewicz defined traces as an algebraic model of finite concurrent processes. In order to modelize non-terminating processes a good notion of infinite trace was needed, which finally led to the notion of complex trace. For complex traces an associative concatenation and an ω-iteration are defined.

This paper defines and investigates rational and recognizable complex trace languages. We prove various closure results such as the closure under boolean operations (for recognizable languages), concatenation, and left and right quotients by recognizable sets. Then we study sufficient conditions ensuring the recognizability of the finite and infinite iterations of complex trace languages. We introduce a generalization of the notion of concurrent iteration which leads to the main result of the paper: the generalization of Kleene's and Ochmanski's theorems to complex trace languages.

0 Introduction

The concept of traces has been introduced by A. Mazurkiewicz (Mazurkiewicz, 1977) as a suitable semantics for non-sequential processes. Let us refer, for instance, surveys

*This research has been supported by the ESPRIT Basic Research Actions No. 6317 ASMICS II and No. 6067 CALIBAN.

(Mazurkiewicz, 1987; Aalbersberg and Rozenberg, 1988; Perrin, 1989) or the monograph (Diekert, 1990). Also, in these references, an extensive bibliography on the subject is given.

There are at least two possible ways to see a trace. A trace can be considered as the set of all possible sequential observations of a concurrent process. The semantics of each sequential observation is given, in a classical way, by a finite word. Hence, a trace is an equivalence class of words. From a different viewpoint, we may see a trace as a finite labeled acyclic graph where edges represent the (causal) dependency of actions. Such a graph is called a dependence graph and the semantics is that an execution has to respect the induced partial order. In this way, the process is modeled as a labeled partial order with an explicit description of concurrency.

Whatever the approach chosen, there is a natural definition of the concatenation of two traces and the set of traces forms a monoid. In fact, this monoid has been introduced and studied independently by Cartier and Foata in combinatorics (Cartier and Foata, 1969).

Recognizable languages describe the behavior of finite state systems and hence form one of the basic families of trace languages. For trace monoids, this family is closed under boolean operations and concatenation, (Fliess, 1974; Cori and Perrin, 1985), but it turns out that a trace language T may be recognizable whereas T^* is not. Hence, in general, the family of recognizable trace languages is strictly included in the family of rational trace languages which is the smallest family of trace languages containing the finite languages and which is closed under union, concatenation, and star-iteration. In order to generalize Kleene's theorem to trace languages it was then necessary to find a new operation instead of star-iteration. In a first step, Métivier and Ochmanski, (Métivier, 1986; Ochmanski, 1985), proved independently that T^* is recognizable if T is recognizable and consists of connected traces, only (this result was also proved by Clerbout and Latteux in the more general framework of semi-commutations (Clerbout and Latteux, 1987)). This led Ochmanski to introduce a concurrent version of the star-iteration, called the c-star in the following. The c-star of T is the usual star taken over all connected components of traces in T. Replacing simply the star by the c-star operation, Ochmanski obtains the equality of recognizable and c-rational trace languages (Ochmanski, 1985). Let us mention that the search for sufficient conditions ensuring the recognizability of T^* is an ongoing work, (Sakarovitch, 1987; Ochmanski, 1990; Métivier and Rozoy, 1991; Gastin et al., 1992).

In order to describe non-sequential processes which never terminate, e.g. distributed operating systems, the notion of infinite trace was needed. It seems that the first explicit definition of infinite traces was given by Mazurkiewicz (Mazurkiewicz, 1987). He defines an (infinite) trace as an (infinite) prefix closed directed subset of finite traces. This is the same as an (infinite) dependence graph where each vertex has finitely many predecessors only, (Mazurkiewicz, 1987, Thm.13). In the following these objects are called here *real traces*. At least implicitly, real traces occur already in (Flé and Roucairol, 1982), but a systematic study began only recently and several papers are devoted to this subject, (Bonizzoni et al., 1990; Kwiatkowska, 1990; Gastin, 1990; Gastin, 1991; Gastin and Rozoy, 1991). One can show that there can be no convenient associative concatenation on real traces, contrary to the set of all (say countable) dependence graphs. However, since the complement of real traces in the monoid of all dependence graphs forms an ideal, one can identify all non-real dependence graphs with a single zero element. In this way, Gastin obtains a monoid which consists of all real traces with only one additional zero element. In this monoid, rational and recognizable languages can be defined in a natural way, (Gastin, 1990). These families are closed under the usual operations, but similarly to the case of finite traces, recognizable sets are not closed neither under star-iteration nor under ω-iteration. In (Gastin et al., 1991) Gastin, Petit and Zielonka gave sufficient conditions which ensure the recognizability of T^* and T^ω. However, substantially they used some finiteness properties due to simplifications by the concatenation with zero.

One of the main drawback of this concatenation is that $a^\omega a = 0 \neq a^\omega$, contrary to the word case. Diekert proposed another solution to the problem of concatenation and defined the notion of complex trace (Diekert, 1991). A possible way to see a complex trace is to start with an arbitrary dependence graph. From this graph we memorize in a first component its maximal real prefix and in a second component we memorize those letters which depend on actions occurring either infinitely often or in the transfinite part of the graph. Thus, a complex trace is simply a real trace together with a second component which is some finite alphabetic information. Therefore, complex traces are slightly less abstract than real traces. The advantage is that the concatenation and ω-iteration are fully defined for complex traces. In particular we have $a^\omega a = a^\omega$ for all letters a. Furthermore, in the special case of a full (empty respectively) dependence relation we just reobtain the usual construct of finite or infinite words (vectors respectively).

The aim of this paper is to define in a proper way rational and recognizable complex trace languages and to investigate their basic properties. We show various closure properties such as closure under boolean operations (for recognizable languages), concatenation, and left and right quotients by recognizable sets. Then we give sufficient conditions ensuring the recognizability of T^* and T^ω. Contrary to the concatenation with zero, in the complex calculation of T^* (T^ω respectively) an unbounded (infinite respectively) number of factors of infinite traces may be relevant. Therefore, we have to develop new techniques, which help to analyze these iterated products. In a next step we introduce the concurrent iterations c-$*$ and c-ω for complex trace languages. For this we define connected components of complex traces and the c-iterations are the usual iterations over all connected components of elements of a given language. This leads to the family of c-rational complex trace languages. Our main result is the generalization of Kleene's and Ochmanski's theorems to complex trace languages. It states that c-rational languages are the same as recognizable languages. This result is a proper generalization of Kleene's theorem for finite words, Büchi's theorem for infinite words and Ochmanski's theorem for finite traces.

The paper is organized as follows: In Section 1 we recall some definitions and facts about dependence graphs, real and complex traces, and rational and recognizable languages. The section on rational and recognizable real trace languages (Section 2) contains all basic material which is crucial for the following. This section contains new results as well as previously known results where, however, different proofs are presented in most cases. The corresponding section for complex trace languages (Section 3) generalizes the results on real traces to complex traces. In the final section (Section 4) we define connected components of complex traces and the concurrent iteration which leads to our main result.

An Extended Abstract of a preliminary version of this paper appeared at MFCS'91, (Diekert et al., 1991).

1 Preliminaries

1.1 Dependence graphs

Let (X, D) be a finite *dependence alphabet*, i.e., X is a finite alphabet with a reflexive and symmetric relation $D \subseteq X \times X$ which is called the *dependence relation*. The complement $I = X \times X \setminus D$ is called the *independence relation*. The quotient monoid $\mathbb{M}(X, D) = X^*/\{ab = ba \mid (a, b) \in I\}$ is the monoid of (finite) *traces*.

It is possible to introduce infinite traces using an equivalence relation on infinite words, the definition of infinite traces is however more natural using dependence graphs. A *dependence graph* (over (X, D)) is (an isomorphism class of) a labeled graph $[V, E, \lambda]$ where V is a countable set of vertices, $E \subseteq V \times V$ is the set of arcs, $\lambda : V \to X$ is the labeling and it holds

- $[V, E]$ is acyclic : $\forall x, y \in V$,

$$(x, y) \in E^* \text{ and } (y, x) \in E^* \Longrightarrow x = y$$

- edges are between dependent vertices : $\forall x, y \in V$,

$$(\lambda(x), \lambda(y)) \in D \Longleftrightarrow x = y \text{ or } (x, y) \in E \text{ or } (y, x) \in E$$

- the induced partial order (V, E^*) is well-founded : there does not exist an infinite sequence $(x_i) \subseteq V$ such that $(x_{i+1}, x_i) \in E$ for all i.

It is also possible to define traces using labeled partial orders. When the dependence alphabet is known, a dependence graph may be identified with a labeled partial order. However, when the dependence alphabet is not known, dependence graphs are more precise.

The set of dependence graphs is denoted by $\mathbb{G}(X, D)$. Note that, in a dependence graph, any subset of vertices with the same label is well-ordered. This allows to think of dependence graphs by standard representation where the vertices are pairs (a, i) with $a \in X$ and i is a countable ordinal. The restriction to countable sets is not essential here, we simply need any upper bound on the cardinality in order to ensure that $\mathbb{G}(X, D)$ is a set.

For $g = [V, E, \lambda] \in \mathbb{G}(X, D)$ the cardinality of V is called the length of g, denoted by $|g|$. For $U \subseteq V$, let $\downarrow U = \{p \in V \mid (p, p') \in E^* \text{ for some } p' \in U\}$ be the downward closure of U. The restriction of g to $\downarrow U$ is a dependence graph. The set of minimal elements of g is $\min(g) = \{p \in V \mid \downarrow p = \{p\}\}$. Since $\min(g) = \downarrow \min(g)$, it may be viewed as a dependence graph which is finite since the alphabet is finite.

The set $\mathbb{G}(X, D)$ is a monoid by the concatenation $[V_1, E_1, \lambda_1] \cdot [V_2, E_2, \lambda_2] = [V, E, \lambda]$, where $[V, E, \lambda]$ is the disjoint union of $[V_1, E_1, \lambda_1]$ and $[V_2, E_2, \lambda_2]$ together with new arcs

(p_1, p_2) for all $p_1 \in V_1$, $p_2 \in V_2$ such that $(\lambda_1(p_1), \lambda_2(p_2)) \in D$. The neutral element is the empty graph $1 = [\emptyset, \emptyset, \emptyset]$.

The concatenation generalizes immediately to an infinite product. Let (g_i) be any sequence of dependence graphs, then $g = g_1 g_2 \ldots \in \mathbb{G}(X, D)$ is defined as the disjoint union of the graphs (g_i) with new arcs from vertices of g_i to vertices of g_j whenever $i < j$ and the vertices have dependent labels. Thus, for any set $L \subseteq \mathbb{G}(X, D)$, the ω-iteration $L^\omega = \{g_1 g_2 \ldots \mid g_i \in L \text{ for } i \geq 1\} \subseteq \mathbb{G}(X, D)$ is defined. (In fact, we can define in the same way L^α for any countable ordinal α.)

Using these products, we define the canonical mapping φ from the free monoid of finite and infinite words X^∞ into the monoid of dependence graphs $\mathbb{G}(X, D)$ by $\varphi(a) = [\{p\}, \emptyset, p \mapsto a]$ for all $a \in X$ and $\varphi(a_1 a_2 \ldots) = \varphi(a_1)\varphi(a_2) \ldots$ for all word $a_1 a_2 \ldots \in X^\infty$. In other words, the dependence graph $\varphi(a_1 a_2 \ldots)$ has vertices $p_1, p_2, \ldots$ labeled by $a_1, a_2, \ldots$ and edges from p_i to p_j if and only if $i < j$ and $(a_i, a_j) \in D$. The congruence induced by φ on X^* is exactly the congruence generated by $\{ab = ba \mid (a, b) \in I\}$. Therefore, the monoid of finite traces $\mathbb{M}(X, D)$ can be identified with the submonoid of finite dependence graphs and we have $\mathbb{M}(X, D) \subseteq \mathbb{G}(X, D)$.

The image $\varphi(X^\infty) \subseteq \mathbb{G}(X, D)$ is called the set of *real traces* and is denoted by $\mathbb{R}(X, D)$. The set of real traces can be characterized as those dependence graphs where every vertex p has a finite downward closure $\downarrow p$. This is a proper subset of $\mathbb{G}(X, D)$.

For instance, let (X, D) be the dependence alphabet represented by the graph $a — b — c — d — e$. The graph f_3 of Figure 1 is a real one whereas the graphs f_1, f_2 and f_4 are non-real graphs. Since a dependence graph is an isomorphism class of graphs, we have only written the labels of the vertices. Still in order to simplify the pictures, we have not necessarily drawn the edges which can be obtained by composition of other edges.

The main disadvantage of $\mathbb{R}(X, D)$ is that there is a convenient notion of concatenation only if D is transitive, i.e., when $\mathbb{M}(X, D) = X_1^* \times \cdots \times X_k^*$ is a direct product of free monoids. In this case, $\mathbb{R}(X, D) = X_1^\infty \times \cdots \times X_k^\infty$. However, even in this case, the concatenation will be different from the (natural) concatenation of $\mathbb{G}(X, D)$. In fact, (for $X \neq \emptyset$) the set $\mathbb{R}(X, D)$ is never a submonoid of $\mathbb{G}(X, D)$. Consider any pair $(a, b) \in D$. Then $a^\omega b \in \mathbb{G}(X, D)$ is not real, since the vertex with label b depends on infinitely many vertices. In particular, the canonical mapping $\varphi : X^\infty \to \mathbb{G}(X, D)$ is not a morphism. It neither commutes with the concatenation nor with the ω-iteration, in general. More precisely, let $L, K \subseteq X^\infty$ then we have $\varphi(L \cdot K) \neq \varphi(L) \cdot \varphi(K)$ and $\varphi(L^\omega) \neq \varphi(L)^\omega$ as soon as L contains an infinite word and K contains a nonempty word.

1.2 Complex traces

We recall briefly in this section the notion of complex traces and some properties that will be needed throughout this paper. The reader is refered to the original paper (Diekert, 1991) for more details.

Every dependence graph $g = [V, E, \lambda] \in \mathbb{G}(X, D)$ splits into its *real part* $Re(g) = \{p \in V \mid \downarrow p \text{ is finite}\}$ and its *transfinite part* $Tr(g) = \{p \in V \mid \downarrow p \text{ is infinite}\}$. Of course, $Re(g)$, $Tr(g)$ are viewed as dependence graphs by restricting E and λ correspondingly. Note that for all $g \in \mathbb{G}(X, D)$ it holds $g = Re(g) \cdot Tr(g)$. The mapping $Re : \mathbb{G}(X, D) \to \mathbb{R}(X, D)$

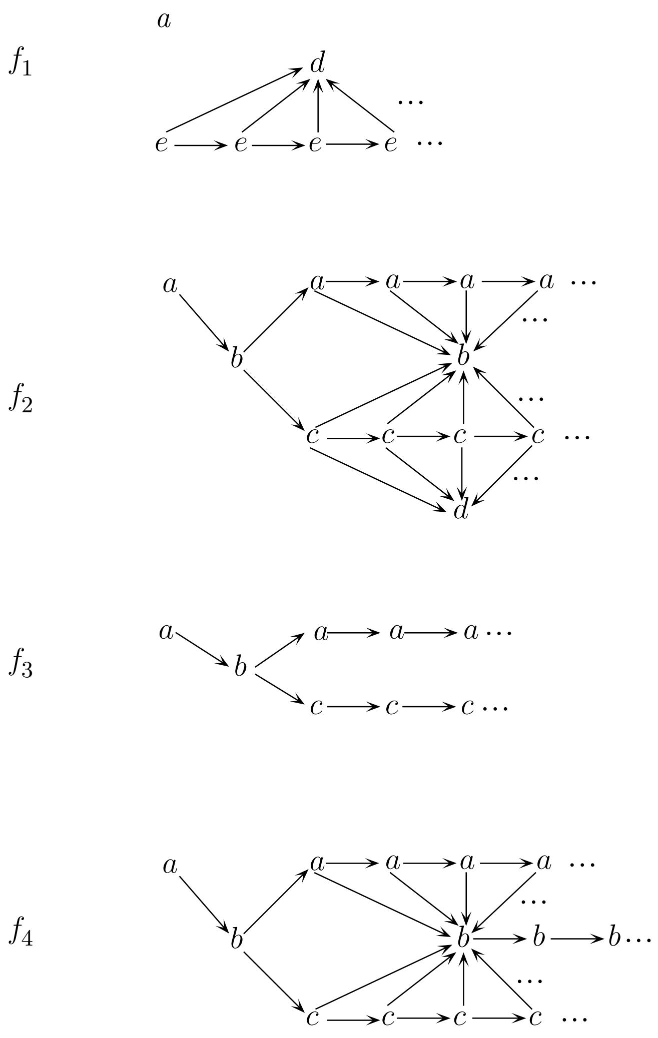

Figure 1: Some dependence graphs

does not define a congruence of $\mathbb{G}(X,D)$. However, we can define the coarsest congruence of $\mathbb{G}(X,D)$ such that any two congruent dependence graphs g and g' verify $Re(g) = Re(g')$. Dependence graphs falling in the same congruence class are called *practically indistinguishable* and the quotient of $\mathbb{G}(X,D)$ by this congruence is the monoid of *complex traces* $\mathbb{C}(X,D)$, (Diekert, 1991).

In order to give an explicit description of $\mathbb{C}(X,D)$ we need a few more notations. For a dependence graph $g \in \mathbb{G}(X,D)$, the *alphabet* of g is the set $\mathrm{alph}(g) = \lambda(V)$ of letters occurring in g. The *alphabet at infinity* of g is the set $\mathrm{alphinf}(g) = \{a \in X \mid \lambda^{-1}(a) \text{ is infinite}\} \cup \mathrm{alph}(Tr(g))$. For a subset $A \subseteq X$ we denote the set of letters depending on A by $D(A) = \{b \in X \mid \exists a \in A : (a,b) \in D\}$, and the set of letters independent of A by $I(A) = X \setminus D(A) = \{b \in X \mid \forall a \in A : (a,b) \in I\}$. Finally, for a dependence graph $g \in \mathbb{G}(X,D)$, we define its *imaginary part* $Im(g)$ by $Im(g) = D(\mathrm{alphinf}(g))$. Intuitively, letters of $\mathrm{Im}(g)$ are those that vanish in the real part whereas others cross infinity. More precisely, $\mathrm{Im}(g) = \{a \in X \mid \mathrm{Re}(ga) = \mathrm{Re}(g)\}$

Using these notations, the coarsest congruence of $\mathbb{G}(X,D)$ which respects real parts admits the following characterization. Two dependence graphs g and g' are congruent (or practically indistinguishable) if and only if $Re(g) = Re(g')$ and $Im(g) = Im(g')$. Therefore, a complex trace is a pair $(Re(g), Im(g))$ for some dependence graph $g \in \mathbb{G}(X,D)$. Often, we denote a complex trace by $(r, D(A))$ where $r \in \mathbb{R}(X,D)$ and $A \subseteq X$. Note that the real part r is uniquely defined by the complex trace whereas the subset $A \subseteq X$ is not known, in general. (It is only $D(A)$ which is given by the complex trace.)

For instance, let $(X,D) = a \text{ --- } b \text{ --- } c \text{ --- } d \text{ --- } e$. Let f_1, f_2, f_3 and f_4 be the dependence graphs represented in Figure 1. The real parts, transfinite parts, alphabets at infinity and imaginary parts of these dependence graphs are given below.

	Re	Tr	alphinf	Im
f_1	ae^ω	d	$\{d,e\}$	$\{c,d,e\}$
f_2	$ab(ac)^\omega$	bd	$\{a,b,c,d\}$	$\{a,b,c,d,e\}$
f_3	$ab(ac)^\omega$	$\mathbf{1}$	$\{a,c\}$	$\{a,b,c,d\}$
f_4	$ab(ac)^\omega$	b^ω	$\{a,b,c\}$	$\{a,b,c,d\}$

Hence, the dependence graphs f_1 and f_2 represent respectively the complex traces $(ae^\omega, D(\{d,e\}))$ and $(ab(ac)^\omega, D(\{a,b,c,d\}))$. The real dependence graph f_3 and the non-real one f_4 represent the same complex trace $(ab(ac)^\omega, D(\{a,c\})) = (ab(ac)^\omega, D(\{a,b,c\}))$.

The concatenation of $\mathbb{C}(X,D)$ is inherited from $\mathbb{G}(X,D)$, but of course it is convenient to have an explicit formula. For this purpose, we introduce the μ-notation. Let $g \in \mathbb{G}(X,D)$ be a dependence graph and $A \subseteq X$ be any subset. Then $\mu_A(g) \in \mathbb{R}(X,D)$ is the maximal real prefix of g containing letters from $I(A)$, only. Thus, for $g = [V,E,\lambda]$, the prefix $\mu_A(g)$ is the restriction of g to $\{p \in V \mid \downarrow p \text{ is finite and } \mathrm{alph}(\downarrow p) \subseteq I(A)\}$. Note that, for $g \in \mathbb{G}(X,D)$ and $A \subseteq X$, we have $\mu_A(g) = \mu_A(Re(g))$. Hence the μ-notation is also well-defined for complex traces. Since $\mathbb{G}(X,D)$ is left-cancellative, (Diekert, 1991), there is a unique dependence graph $\mathrm{Suff}_A(g)$ such that $g = \mu_A(g) \cdot \mathrm{Suff}_A(g)$. With these notations the concatenation in $\mathbb{C}(X,D)$ becomes (Diekert, 1991):

$$(r, D(A)) \cdot (s, D(B)) = (r \cdot \mu_A(s), D(A \cup B \cup \mathrm{alph}(\mathrm{Suff}_A(s))))$$

This formula is of first importance for the calculus on complex traces and will be used throughout.

For instance, with $(X,D) = a — b — c — d — e$, it holds $\mu_{\{d,e\}}(ab(ac)^\omega) = aba^\omega$ and $\mathrm{Suff}_{\{d,e\}}(ab(ac)^\omega) = c^\omega$. Hence

$$\begin{aligned}(ae^\omega, D(\{d,e\})) \cdot (ab(ac)^\omega, D(\{a,c\})) &= (a^2b(ae)^\omega, D(\{d,e\}) \cup D(\{a,c\}) \cup D(\{c\})) \\ &= (a^2b(ae)^\omega, D(\{a,c,d,e\}))\end{aligned}$$

It turns out that the coarsest congruence on $\mathbb{G}(X,D)$ which respects real parts is also a "congruence" for the ω-product on $\mathbb{G}(X,D)$. Therefore, the ω-iteration L^ω of a complex trace language L is well-defined. Note that, following the general definition of the ω-product for dependence graphs, we have $L^\omega = L^* \cup (L \setminus \{1\})^\omega$ if $1 \in L$.

In the remainder, we identify a real trace r with the complex trace $(r, D(\mathrm{alphinf}(r))) = (r, Im(r))$. Thus, we have $\mathbb{M}(X,D) \subseteq \mathbb{R}(X,D) \subseteq \mathbb{C}(X,D)$ and $\mathbb{C}(X,D)$ is a quotient monoid of $\mathbb{G}(X,D)$. For a subset $A \subseteq X$ and a language $L \subseteq \mathbb{C}(X,D)$ ($L \subseteq \mathbb{R}(X,D)$ respectively, $L \subseteq \mathbb{G}(X,D)$ respectively) we define $L_A = \{x \in L \mid Im(x) = D(A)\}$. For instance,

$$\begin{aligned}\mathbb{R}(X,D)_A &= \{r \in \mathbb{R}(X,D) \mid D(\mathrm{alphinf}(r)) = D(A)\} \\ &= \cup_{\{B \subseteq X \mid D(B) = D(A)\}}(\mathbb{M}(X,D) \cdot \{s \in \mathbb{M}(X,D) \mid \mathrm{alph}(s) = B\}^\omega)\end{aligned}$$

Note that $\mathbb{R}(X,D)_\emptyset = \mathbb{C}(X,D)_\emptyset = \mathbb{G}(X,D)_\emptyset = \mathbb{M}(X,D)$.

The link between real and complex trace languages can be done using concatenations by finite shift-traces. A finite trace $s_{A,B}$ is called a *shift trace* from $D(A)$ to $D(B)$ if for some complex trace $(r, D(A))$ we have $(r, D(A)) \cdot s_{A,B} = (r, D(B))$. In this case, we have $(t, D(A)) \cdot s_{A,B} = (t, D(B))$ for all complex traces $(t, D(A))$. A shift trace $s_{A,B}$ satisfies $\mu_A(s_{A,B}) = 1$ (i.e., $\min(s_{A,B}) \subseteq D(A)$) and $D(A \cup \mathrm{alph}(s_{A,B})) = D(B)$. Shift traces from $D(A)$ to $D(B)$ do not exist for all pairs (A,B). The necessary and sufficient condition is that there exist some $k \geq 0$ and a sequence of letters $a_1, \ldots, a_k$ such that $a_i \in D(A \cup \{a_1, \ldots, a_{i-1}\})$ for $1 \leq i \leq k$ and $D(B) = D(A \cup \{a_1, \ldots, a_k\})$. We simply write A shift B in this case. If A shift B then some shift trace $s_{A,B}$ can be chosen of length at most $|D(B)|$.

For instance, with $(X,D) = a — b — c — d — e$, it holds $(ab(ac)^\omega, D(\{a,b,c,d,e\})) = (ab(ac)^\omega, D(\{a,c\})) \cdot (bd, \emptyset) = ab(ac)^\omega \cdot bd$.

It is easy to see that every complex trace language $L \subseteq \mathbb{C}(X,D)$ can be written as a finite union of real trace languages concatenated by shift traces:

$$L = \bigcup_{A \text{ shift } B} (Re(L_B) \cap \mathbb{R}(X,D)_A) \cdot s_{A,B} \tag{1}$$

In particular, any complex trace is the product of a real trace by a finite (shift-)trace. Since a real trace is the product of at most ω finite traces, a complex trace can be written as a product of at most $\omega + 1$ finite traces, we have:

$$\mathbb{C}(X,D) = \mathbb{M}(X,D)^{\omega+1} \tag{2}$$

Let us point out that the word monoid X^∞ with right-absorbent concatenation $x \cdot y = x$ for $x \in X^\omega$ is just the special case of the complex trace monoid $\mathbb{C}(X,D)$ for a full dependence relation $D = X \times X$. In fact, if the independence relation is empty then the imaginary part is redundant. For a word x we have $Im(x) = \emptyset$ if x is finite and $Im(x) = X$ otherwise (i.e., if x is an infinite word). Furthermore, the partition $X^\infty = X^* \cup X^\omega$ is the partition of X^∞ into $(X^\infty)_\emptyset$ and $(X^\infty)_X$.

In the remainder, we will study rational and recognizable complex trace languages. As it becomes clear from Equation (1), the overall strategy is to transfer results from real trace languages. Since $\mathbb{R}(X, D) = \varphi(X^\infty)$, a basic technique used here to obtain results on real traces is to work with representing words. The next section briefly recalls some well-known properties of word languages.

1.3 Recognizable and rational word languages

We have (at least) two possibilities to define regular subsets of the monoid of finite and infinite words X^∞. We can use either rational expressions or finite state acceptors. This leads to the families of rational and recognizable languages respectively (see (Eilenberg, 1974; Thomas, 1990a; Perrin and Pin, 1991) for complete studies of these families).

The family of rational word languages, $\mathrm{Rat}(X^\infty)$, is defined as the smallest family of subsets of X^∞ which contains all finite languages of finite words and which is closed under the operations union, concatenation, $*$-iteration, and ω-iteration. From the definition of the concatenation in X^∞, it is clear that we may restrict the operations of concatenation $L \cdot K$, of Kleene-$*$ L^*, and of ω-iteration L^ω to the case of languages $L \subseteq X^*$. Thus, we are never forced to use the right-absorbent concatenation $x \cdot y = x$ with $x \in X^\omega$.

We define recognizable word languages using non-deterministic Büchi automata. There are several equivalent possibilities to give acceptance conditions. For technical reasons we use here acceptance by transitions. This allows in particular to construct automata which accept languages from X^∞ without separation of the finitary part. Moreover, this acceptance type is very convenient for our purposes since from the set of transitions which occur infinitely often in a path, we can deduce directly the alphabet at infinity of its label. To be precise, we define a Büchi automaton to be a tuple (Q, δ, q_0, Δ) where Q is the finite set of states, $\delta \subseteq Q \times (X \cup \{1\}) \times Q$ is a finite set of labeled arcs called transitions, $q_0 \in Q$ is the initial state, and $\Delta \subseteq \delta$ is the set of final transitions. An infinite path is a sequence $q_0, x_1, q_1, x_2, q_2, \ldots$ where $(q_{i-1}, x_i, q_i) \in \delta$ for all $i > 0$ and its label is the word $x_1 x_2 \cdots \in X^\infty$. A finite or infinite word x is accepted if there exists an infinite path labeled by x, starting in q_0 and repeating some transition from Δ infinitely often. Note that, finite words can be accepted by infinite loops of 1-transitions. A language $L \subseteq X^\infty$ is called *recognizable* if it is accepted by some Büchi automaton.

Another equivalent definition for recognizable languages is to use recognizing morphisms. Let $\eta : X^* \to S$ be a morphism to a finite monoid S. We say that η *recognizes* a language $L \subseteq X^\infty$ if for any finite or infinite sequence $x_1, x_2, \ldots$ of finite words, we have $x_1 x_2 \ldots \in L$ implies $\eta^{-1}\eta(x_1)\eta^{-1}\eta(x_2)\ldots \subseteq L$. Using Ramsey factorization (see below), one can show that any recognizable language can be written as a finite union

$$L = \bigcup_{(s,e) \in P} X_s X_e^\omega \tag{3}$$

where $X_s = \eta^{-1}(s)$, $X_e = \eta^{-1}(e)$ and $P = \{(s, e) \in S \times S \mid se = s,\ e^2 = e,\ X_s X_e^\omega \cap L \neq \emptyset\}$.

Büchi's classical theorem asserts that for X^∞ the families of rational languages and of recognizable languages coincide.

1.4 Ramsey factorization

Let $\eta : M \longrightarrow S$ be a morphism from an arbitrary monoid M to a finite monoid S and let $u_0, u_1, \ldots$ be an infinite sequence of elements of M. There exist an infinite increasing sequence of integers $i_1 < i_2 < \ldots$ and $s, e \in S$ such that $se = s$, $e^2 = e$, $\eta(u_0 \ldots u_{i_1}) = s$ and $\eta(u_{1+i_j} \ldots u_{i_k}) = e$ for all $1 \leq j < k$. This is called a *Ramsey factorization* of the infinite sequence $u_0, u_1, \ldots$.

The existence of a Ramsey factorization is one of the fundamental principles used in this paper. Therefore, and for the sake of completeness, we include a simple and direct proof of this result inspired from (Perrin and Pin, 1991). It is also a trivial application of Ramsey's Theorem (Graham, 1981).

Let $V = \{(i,j) \in \mathbb{N}^2 \mid i < j\}$. We consider the mapping $c : V \longrightarrow S$ defined by $c(i,j) = \eta(u_{i+1} \ldots u_j)$. We define inductively a sequence of infinite sets of integers $N_0 \supset N_1 \supset N_2 \ldots$ and a sequence of integers $n_0 < n_1 < n_2 \ldots$. First, we set $N_0 = \mathbb{N}$. Assume that N_p is already defined and let $n_p = \min(N_p)$. Since S is finite and N_p is infinite, there exist $s_p \in S$ and an infinite subset $N_{p+1} \subset N_p$ such that $c(n_p, m) = s_p$ for all $m \in N_{p+1}$. Finally, since S is finite, there exist $e \in S$ and an infinite sequence $p_0 < p_1 < p_2 \ldots$ such that $s_{p_j} = e$ for all $j \geq 0$. Then, for all $j \geq 1$, we set $i_j = n_{p_j}$. For all $1 \leq j < k$ we have $\eta(u_{1+i_j} \ldots u_{i_k}) = c(i_j, i_k) = s_{p_j} = e$. Note that e is idempotent since $e^2 = s_{p_0} s_{p_1} = c(n_{p_0}, n_{p_1}) c(n_{p_1}, n_{p_2}) = c(n_{p_0}, n_{p_2}) = s_{p_0} = e$. Moreover, if we set $s = \eta(u_0 \ldots u_{i_1})$ we obtain $s = \eta(u_0 \ldots u_{n_{p_0}}) c(n_{p_0}, n_{p_1}) = \eta(u_0 \ldots u_{n_{p_0}}) e = \eta(u_0 \ldots u_{n_{p_0}}) ee = se$ and the result is proved.

2 Rational and recognizable real trace languages

This section is devoted to the study of rational and recognizable languages of real traces. It contains new results as well as previously known results where, however, different proofs are presented in most cases.

Several characterizations of rational real trace languages (in particular a kind of normal form) are proposed in Proposition 2.1. Rational languages of real traces are also shown to be closed under quotients by recognizable languages (Proposition 2.14).

Characterizations of recognizable languages of real traces in terms of recognizing morphisms and syntactic congruence are presented in Proposition 2.3. One of the main result of this section states that if a recognizable language L of finite traces satisfies $L = L^+$, then L^ω is still recognizable (Theorem 2.7). Using in particular this result and the closure of recognizable languages of real traces under concatenation (Proposition 2.10) a kind of normal form is proposed in Proposition 2.11. Finally, recognizable languages of real traces are closed under quotients (Proposition 2.13).

2.1 Rational languages

Recall that in a general monoid M, the family $\mathrm{Rat}(M)$ is the least family which contains the finite sets and which is closed under union, product and Kleene's iteration. This definition applies in particular for the free monoid X^* and the finitary trace monoid $\mathbb{M}(X, D)$. In order to define the family $\mathrm{Rat}(X^\infty)$, we start with finite sets of finite words and the closure under ω-iteration is required, too. Since there is no concatenation defined for real traces in general, we use restricted operations, only.

Definition: *The family of rational real trace languages,* $\mathrm{Rat}(\mathbb{R}(X, D))$, *is the smallest family satisfying the following conditions:*

i) Every finite set of finite traces is rational.

ii) If $K, L \subseteq \mathbb{R}(X, D)$ are rational, then the union $K \cup L$ is rational.

iii) If $K, L \subseteq \mathbb{R}(X, D)$ are rational and if in addition $K \subseteq \mathbb{M}(X, D)$, then $K \cdot L$, K^*, and K^ω are rational.

This definition is consistent with the definitions of $\mathrm{Rat}(X^\infty)$ and $\mathrm{Rat}(\mathbb{M}(X, D))$ since $\mathrm{Rat}(\mathbb{R}(X, D)) = \mathrm{Rat}(X^\infty)$ when $D = X \times X$ is the full dependence relation and for $L \subseteq \mathbb{M}(X, D)$ we have $L \in \mathrm{Rat}(\mathbb{R}(X, D))$ if and only if $L \in \mathrm{Rat}(\mathbb{M}(X, D))$.

Note that the canonical mapping $\varphi : X^\infty \to \mathbb{R}(X, D)$ (defined in Section 1.1) is not a morphism for the usual concatenation on infinite words, but it commutes with these restricted operations. This is used in the next proposition which gives several equivalent characterizations of the family $\mathrm{Rat}(\mathbb{R}(X, D))$.

Proposition 2.1 ((Gastin, 1991)) *Let $L \subseteq \mathbb{R}(X, D)$ be any real trace language. Then the following assertions are equivalent.*

i) The language L is rational, i.e., $L \in \mathrm{Rat}(\mathbb{R}(X, D))$

ii) The language L is the image of a rational word language, i.e., $L = \varphi(L')$ for some $L' \in \mathrm{Rat}(X^\infty)$.

iii) The language L can be written as a finite union

$$L = \bigcup_{1 \le i \le k} M_i N_i^\omega$$

where M_i, $N_i \subseteq \mathbb{M}(X, D)$ are rational trace languages for all $1 \le i \le k$.

Proof: i) $\Rightarrow$ ii): First, assume that L is a finite set of finite traces. We have $L = \varphi(\varphi^{-1}(L))$ and $\varphi^{-1}(L)$ is rational since it is finite. Now, for $K', L' \subseteq X^\infty$ we have $\varphi(K' \cup L') = \varphi(K') \cup \varphi(L')$ and if moreover $K' \subseteq X^*$ then $\varphi(K' \cdot L') = \varphi(K') \cdot \varphi(L')$, $\varphi(K'^*) = \varphi(K')^*$ and $\varphi(K'^\omega) = \varphi(K')^\omega$. Therefore, the result follows by induction on the rational expression defining L.
ii) $\Rightarrow$ iii): Using Equation (3) we obtain $L' = \bigcup_{1 \le i \le k} M'_i \cdot N_i'^\omega$ where $M'_i, N'_i \in \mathrm{Rat}(X^*)$ for $1 \le i \le k$. Since φ commutes with these operations, we obtain $L = \bigcup_{1 \le i \le k} \varphi(M'_i) \cdot \varphi(N'_i)^\omega$. Using again the same property of φ we deduce that the languages $\varphi(M'_i), \varphi(N'_i)$ are rational languages of finite traces, which concludes this part.
iii) $\Rightarrow$ i): trivial. $\square$

2.2 First characterizations of recognizable languages

Recall that we have defined recognizable word languages by Büchi automata. In order to define recognizable real trace languages we will consider automata which accept closed languages. A word language $L \subseteq X^\infty$ is said to be *closed* (with respect to (X, D)) if $L = \varphi^{-1}\varphi(L)$ for the canonical mapping $\varphi : X^\infty \to \mathbb{R}(X, D)$. If a Büchi automaton accepts a closed language then there is no ambiguity in saying which real traces are accepted. Moreover we will see that it is decidable whether a Büchi automaton accepts a closed language or not. This approach leads to the following definition.

Definition: *A language $L \subseteq \mathbb{R}(X, D)$ is called* recognizable *if $\varphi^{-1}(L)$ is a (closed) recognizable word language. The family of recognizable real trace languages is denoted by* $\mathrm{Rec}(\mathbb{R}(X, D))$.

Remark 2.2 Since the family $\mathrm{Rec}(X^\infty)$ is a boolean algebra, it follows directly from this definition that the family of recognizable real trace languages forms a boolean algebra, too. It is closed under union, intersection, and complementation.

Another possible way to define recognizable languages is, exactly as in the word case, by *recognizing morphisms*. Let $\eta : \mathbb{M}(X, D) \to S$ be a morphism to a finite monoid S. We say that η recognizes a language $L \subseteq \mathbb{R}(X, D)$, if for any finite or infinite sequence $(x_i) \subseteq \mathbb{M}(X, D)$, $x_1 x_2 \ldots \in L$ implies $\eta^{-1}\eta(x_1)\eta^{-1}\eta(x_2)\ldots \subseteq L$. We could also use the generalization to trace languages of the syntactic congruence defined by Arnold for word languages (Arnold, 1985). Given $L \subseteq \mathbb{R}(X, D)$, two finite traces x, y are syntactically congruent if and only if:

$$\begin{aligned} \forall u, v \in \mathbb{M}(X, D) &\ : \ u(xv)^\omega \in L \Leftrightarrow u(yv)^\omega \in L \\ \forall u, v, w \in \mathbb{M}(X, D) &\ : \ uxvw^\omega \in L \Leftrightarrow uyvw^\omega \in L \end{aligned}$$

We denote the syntactic congruence of L by $\equiv_L$ and we obtain a canonical morphism $\eta_L : \mathbb{M}(X, D) \to \mathbb{M}(X, D)/\equiv_L$. Similarly to the word case, in general, neither $\mathbb{M}(X, D)/\equiv_L$ is finite nor η_L recognizes L. This is true however if L is recognizable. We have the following proposition, stating that all approaches are equivalent.

Proposition 2.3 ((Gastin, 1991)) *Let $L \subseteq \mathbb{R}(X, D)$ be a real trace language. Then, the following assertions are equivalent.*

i) The language L is recognizable.

ii) There exists a morphism η from $\mathbb{M}(X, D)$ to a finite monoid S recognizing L.

iii) The syntactic congruence $\equiv_L$ is of finite index in $\mathbb{M}(X, D)$ and the syntactic morphism η_L recognizes L.

From Propositions 2.1 and 2.3, it is clear that every recognizable language is rational. The converse holds for strings by Büchi's Theorem. However, as soon as there is a pair $(a, b) \in I$ of independent letters, we find a rational language, e.g., $L = \varphi(ab)^* \subseteq \mathbb{M}(X, D)$, which is rational but not recognizable (since $\varphi^{-1}(L) = \{w \in \{a,b\}^* \mid |w|_a = |w|_b\}$ is not a recognizable word language).

An example of an ω-language $L \subseteq \mathbb{R}(X,D) \setminus \mathbb{M}(X,D)$ which is rational but not recognizable, can be constructed on three letters, say a, b, c where there is at least one pair of independent letters, say $(a,b) \in I$. Then the rational language $L = \varphi(ab)^* \cdot \varphi(c)^\omega$ is not recognizable since the projection of $\varphi^{-1}(L)$ to $\{a,b\}^\infty$ is $\{w \in \{a,b\}^* \mid |w|_a = |w|_b\}$.

We leave it as an exercise to the reader to show that no such two letters example exists.

Rational trace languages are not closed under intersection, in general. For example, let $(X,D) = a \quad b — c$ then $\varphi(ab)^*\varphi(c)^* \cap \varphi(b)^*\varphi(ac)^* = \{\varphi(a^n b^n c^n) \mid n \geq 0\}$, which is not rational since the projection to $\{b,c\}^*$ yields the non rational language $\{b^n c^n \mid n \geq 0\} \subseteq \{b,c\}^*$. However we have the following observation. It is stated as a lemma since we will use it throughout.

Lemma 2.4 *Let $K \subseteq \mathbb{R}(X,D)$ be rational and $L \subseteq \mathbb{R}(X,D)$ be recognizable. Then the intersection $K \cap L$ is rational.*

Proof: Let $K', L' \subseteq X^\infty$ be recognizable word languages such that $\varphi(K') = K$, $\varphi(L') = L$ and L' is closed. Then $K' \cap L'$ is a recognizable word language and we have $\varphi(K' \cap L') = K \cap L$. Hence, by Proposition 2.1, $K \cap L$ is rational. □

2.3 Iteration

The following combinatorial lemma becomes important below, in particular to study the closure of recognizable languages under iteration.

Lemma 2.5 *Let $t = t_1 t_2 \ldots$, $z = z_1 z_2 \ldots \in X^\infty$ be two factorizations for $t, z \in X^\infty$ with $t_i, z_i \in X^*$. Then we have $\varphi(t) = \varphi(z) \in \mathbb{R}(X,D)$ if and only if there are sequences of integers $0 = m_0 < m_1 < m_2 < \ldots$, $0 = n_0 < n_1 < n_2 < \ldots$, and finite words $1 = y_0, x_1, y_1, x_2, y_2, \ldots \in X^*$ such that it holds for all $i \geq 1$:*

$$\begin{aligned} \varphi(y_{i-1}x_i) &= \varphi(t_{1+m_{i-1}} \cdots t_{m_i}) \\ \varphi(x_i y_i) &= \varphi(z_{1+n_{i-1}} \cdots z_{n_i}) \end{aligned}$$

A rough picture of the situation described in the lemma above is given in the following picture.

$$\begin{array}{rcllll} \varphi(t) &=& \varphi(t_1 \cdots t_{m_1}) & \varphi(t_{1+m_1} \cdots t_{m_2}) & \varphi(t_{1+m_2} \cdots t_{m_3}) & \ldots \\ &=& \varphi(x_1) & \varphi(y_1 x_2) & \varphi(y_2 x_3) & \ldots \\ &=& \varphi(x_1 y_1) & \varphi(x_2 y_2) & \varphi(x_3 y_3) & \ldots \\ &=& \varphi(z_1 \cdots z_{n_1}) & \varphi(z_{1+n_1} \cdots z_{n_2}) & \varphi(z_{1+n_2} \cdots z_{n_3}) & \ldots \\ &=& \varphi(z) & & & \end{array}$$

Double factorization of two words representing the same real trace

Proof: It is clear that the conditions imply $\varphi(t) = \varphi(y_0 x_1 y_1 x_2 y_2 \ldots) = \varphi(z)$. Therefore it is enough to show the other direction and we may assume $\varphi(t) = \varphi(z)$. Let $y_0 = 1$, $x_1 = t_1$

and $m_1 = 1$. Then we may assume by induction that for some $k \geq 1$ we found indices $0 = m_0 < \ldots < m_k$ and $0 = n_0 < \ldots < n_{k-1}$ and finite words $1 = y_0, x_1, \ldots, y_{k-1}, x_k$ such that

$$\begin{aligned} \varphi(y_{i-1}x_i) &= \varphi(t_{1+m_{i-1}} \ldots t_{m_i}) \quad \text{for } 1 \leq i \leq k \\ \varphi(x_i y_i) &= \varphi(z_{1+n_{i-1}} \ldots z_{n_i}) \quad \text{for } 1 \leq i \leq k-1. \end{aligned}$$

Since $\varphi(y_0 x_1 \ldots y_{k-1} x_k)$ is a finite prefix of $\varphi(t) = \varphi(z)$ and $\varphi(y_0 x_1 \ldots x_{k-1} y_{k-1}) = \varphi(z_1 \ldots z_{n_{k-1}})$ it follows from the left-cancellativity of $\mathbb{R}(X, D)$ that $\varphi(x_k)$ is a finite prefix of $\varphi(z_{1+n_{k-1}} z_{2+n_{k-1}} \ldots)$. Hence we find some $n_k > n_{k-1}$ such that $\varphi(x_k)$ is a prefix of $\varphi(z_{1+n_{k-1}} \ldots z_{n_k})$, i.e., $\varphi(x_k y_k) = \varphi(z_{1+n_{k-1}} \ldots z_{n_k})$ for some $y_k \in X^*$. A symmetric argument yields that for some $m_{k+1} > m_k$ there exists $x_{k+1} \in X^*$ such that $\varphi(y_k x_{k+1}) = \varphi(t_{1+m_k} \ldots t_{m_{k+1}})$. This concludes the proof of the lemma. □

A first application of the lemma above shows that it is decidable whether a recognizable word language is closed by computing its syntactic congruence. If a language $L \subseteq X^\infty$ is closed, then obviously its syntactic congruence $\equiv_L$ verifies $ab \equiv_L ba$ for all $(a, b) \in I$. The converse is easy for languages $L \subseteq X^*$. The next theorem states that the converse is true, in general. The decidability result follows since the syntactic congruence of a recognizable word language is computable (Perrin and Pin, 1991).

Proposition 2.6 *A recognizable language $L \subseteq X^\infty$ is closed (i.e., $L = \varphi^{-1}\varphi(L)$) if and only if we have $ab \equiv_L ba$ for all $(a, b) \in I$.*

Proof: We have to show that $ab \equiv_L ba$ for all $(a, b) \in I$ implies $L = \varphi^{-1}\varphi(L)$. Let $t \in L$ and $z \in X^\infty$ such that $\varphi(t) = \varphi(z)$. Applying Lemma 2.5 to the trivial factorizations $t = a_1 a_2 \ldots$ and $z = b_1 b_2 \ldots$ with $a_i, b_i \in X \cup \{1\}$, we find two new factorizations $t = t_1 t_2 \ldots$ and $z = z_1 z_2 \ldots$ with $t_i, z_i \in X^*$ and a sequence of finite words $1 = y_0, x_1, y_1, x_2, y_2, \ldots \in X^*$ such that $\varphi(y_{i-1}x_i) = \varphi(t_i)$ and $\varphi(x_i y_i) = \varphi(z_i)$ for $i \geq 1$.

Let $\eta_L : X^* \to X^*/\equiv_L$ be the syntactic morphism of L. Since $ab \equiv_L ba$ for all $(a, b) \in I$ we have $\eta(t_i) = \eta(y_{i-1}x_i)$ for $i \geq 1$. Since η recognizes L this implies $y_0 x_1 y_1 x_2 \ldots \in L$. In the same way, $\eta(z_i) = \eta(x_i y_i)$ for $i \geq 1$, whence $z \in L$. □

However, the main interest in Lemma 2.5 results from the fact that it yields a simplified proof for the following result.

Theorem 2.7 ((Gastin et al., 1991)) *Let $L \subseteq \mathbb{M}(X, D)$ be a recognizable trace language such that $L = L^+$. Then $L^\omega \subseteq \mathbb{R}(X, D)$ is recognizable, too.*

Proof: Let (Q, δ, q_0, F) be a classical finite deterministic word automaton recognizing $\varphi^{-1}(L)$ by final states, i.e., $F \subseteq Q$ and a finite path is accepted if its initial state is q_0 and its final state is in F.

We now construct a Büchi automaton recognizing $\varphi^{-1}(L^\omega)$ as follows. The set of states is $Q \times Q \times \mathcal{P}(X)$ and the initial state is $(q_0, q_0, \emptyset)$. For each letter $a \in X$ and state $(q, q', A) \in Q \times Q \times \mathcal{P}(X)$ we define the transition

$$(q, q', A) \xrightarrow{a} (q, \delta(q', a), A \cup \{a\})$$

and if $a \in I(A)$ then we define in addition the transition

$$(q, q', A) \xrightarrow{a} (\delta(q, a), q', A)$$

Furthermore, if $q \in F$, we add a final 1-transition:

$$(q, q', A) \xrightarrow{1} (q', q_0, \emptyset).$$

Such a non-deterministic jump signifies that a member of $\varphi^{-1}(L)$ has been computed in the first component.

Note that, if we have a path $(p, p', A) \xrightarrow{*}^{z} (q, q', B)$ without 1-transition in this automaton then there exist $x, y \in X^*$ such that $\varphi(z) = \varphi(xy)$, $q = \delta(p, x)$, $q' = \delta(p', y)$, $alph(x) \subseteq I(A)$ and $B = A \cup alph(y)$. (The converse is trivially true.)

According to the definition of Section 1.3 this automaton accepts all finite or infinite words which are labelings of infinite paths using infinitely many final 1-transitions. By definition, the accepted language is recognizable. Thus, we have to show only that the accepted language is $\varphi^{-1}(L^\omega)$.

First, let $z \in X^\infty$ be an accepted word. Marking the final transitions in its accepted path, we obtain the factorization:

$$\begin{array}{cccc} & (q_0, q_0, \emptyset) & \xrightarrow{*}^{z_1} & (q_1, q_1', A_1) \\ \xrightarrow{1} & (q_1', q_0, \emptyset) & \xrightarrow{*}^{z_2} & (q_2, q_2', A_2) \\ \xrightarrow{1} & (q_2', q_0, \emptyset) & \xrightarrow{*}^{z_3} & \ldots \end{array}$$

with $z = z_1 z_2 z_3 \ldots$.

Using the remark above, we find a sequence of finite words $x_1, y_1, x_2, y_2, \ldots \in X^*$ such that, with $q_0' = q_0$, we have $\varphi(z_i) = \varphi(x_i y_i)$, $q_i = \delta(q_{i-1}', x_i)$ and $q_i' = \delta(q_0, y_i)$ for all $i \geq 1$. Thus, with $y_0 = 1$, we obtain for all $i \geq 1$ $\delta(q_0, y_{i-1} x_i) = q_i \in F$, i.e., $y_{i-1} x_i \in \varphi^{-1}(L)$. Hence, we have $\varphi(z) = \varphi(x_1 y_1)\varphi(x_2 y_2) \ldots = \varphi(y_0 x_1)\varphi(y_1 x_2) \ldots \in L^\omega$.

For the other direction let $z = b_1 b_2 \ldots \in X^\infty$ be any word in $\varphi^{-1}(L^\omega)$. Then we have $\varphi(z) = \varphi(t)$ for some $t = t_1 t_2 \ldots \in X^\infty$ with $t_i \in \varphi^{-1}(L)$ for all $i \geq 1$. Using Lemma 2.5, we obtain a sequence of finite words $1 = y_0, x_1, y_1, x_2, y_2, \ldots \in X^*$ and two factorizations $z = z_1 z_2 \ldots$ and $t = (t_1 \ldots t_{m_1})(t_{1+m_1} \ldots t_{m_2}) \ldots$ such that for all $i \geq 1$ we have

$$\varphi(z_i) = \varphi(x_i y_i) \text{ and } \varphi(y_{i-1} x_i) = \varphi(t_{1+m_{i-1}} \ldots t_{m_i}) \in L^* = L$$

Therefore, we find an accepting path for z as follows:

$$\begin{array}{cccc} & (q_0, q_0, \emptyset) & \xrightarrow{*}^{z_1} & (\delta(q_0, x_1), \delta(q_0, y_1), \mathrm{alph}(y_1)) \\ \xrightarrow{1} & (\delta(q_0, y_1), q_0, \emptyset) & \xrightarrow{*}^{z_2} & (\delta(q_0, y_1 x_2), \delta(q_0, y_2), \mathrm{alph}(y_2)) \\ \xrightarrow{1} & (\delta(q_0, y_2), q_0, \emptyset) & \xrightarrow{*}^{z_3} & \ldots \end{array}$$

□

A finite trace is called *connected* if it is connected as a dependence graph or what is the same if its alphabet induces a connected subgraph of (X, D). A language of finite traces is called *connected* if all its elements are connected. Due to independent works by Métivier (Métivier, 1986) and Ochmanski (Ochmanski, 1985), it is known that L^* (and hence L^+) is recognizable if L is recognizable and connected. Therefore we have the following corollary which is the main application of Theorem 2.7.

Corollary 2.8 *Let $L \subseteq \mathbb{M}(X, D)$ be recognizable and connected. Then L^ω is recognizable.*

Proof: We have $L^\omega = (L^+)^\omega$. $\square$

2.4 Concatenation and quotients

Note that in $\mathbb{G}(X, D)$, the concatenation xy of two real traces is real if and only if $\mathrm{alphinf}(x) \times \mathrm{alph}(y) \subseteq I$. Therefore, even if the concatenation $K \cdot L$, as a rational operation on real trace languages, has been defined only when $K \subseteq \mathbb{M}(X, D)$, it can clearly be extended to the case where $\mathrm{alphinf}(x) \times \mathrm{alph}(y) \subseteq I$ for all $x \in K$ and $y \in L$. The reason that the rational operation was defined in a more restricted way was to maintain commutation with the canonical mapping $\varphi : X^\infty \to \mathbb{R}(X, D)$. This property is not important for the closure under concatenation of $\mathrm{Rec}(\mathbb{R}(X, D))$. It was proved by Gastin (Gastin, 1991) that the family of recognizable real trace languages is closed under this extension of the real concatenation. The key to the proof of this result is the following lemma which introduces the I-shuffle and for which we give a new proof using non-deterministic Büchi automata.

For the sake of simplicity, we write $(x, y) \in I$ if two words or real traces x and y are fully independent, i.e., if $\mathrm{alph}(x) \times \mathrm{alph}(y) \subseteq I$.

Lemma 2.9 *Let $K', L' \subseteq X^\infty$ be recognizable. Then the following I-shuffle is recognizable, too.*

$$K' \sqcup\!\sqcup_I L' = \{u_1 v_1 u_2 v_2 \ldots \in X^\infty \mid u_1 u_2 \ldots \in K', v_1 v_2 \ldots \in L' \text{ and } (v_i, u_j) \in I \text{ for all } i < j\}$$

Proof: For $i = 1, 2$, let $\mathcal{A}_i = (Q_i, \delta_i, q_{i,0}, \Delta_i)$ be Büchi automata accepting K' and L' respectively. We may assume that for $i = 1, 2$, $(q_i, 1, q_i) \in \delta_i$ for all $q_i \in Q_i$. We construct an automaton $\mathcal{A}$ for $K' \sqcup\!\sqcup_I L'$ as follows. The set of states is $Q_1 \times Q_2 \times \mathcal{P}(X) \times \mathcal{P}(\{1, 2\})$ and the initial state is $(q_{1,0}, q_{2,0}, \emptyset, \emptyset)$. We define simultaneously the transitions of $\mathcal{A}$ and their projections on $\mathcal{A}_1$ and $\mathcal{A}_2$.

- If $t_1 = (q_1, x, q_1') \in \delta_1$ and $\mathrm{alph}(x) \times A \subseteq I$ then

  $$(q_1, q_2, A, B) \xrightarrow{x} (q_1', q_2, A, B')$$

  with $B' = B \cup \{1\}$ if $t_1 \in \Delta_1$ and $B' = B$ otherwise.

  Moreover, the projections of this transition are t_1 and $(q_2, 1, q_2)$ respectively.

- If $t_2 = (q_2, x, q_2') \in \delta_2$ then

  $$(q_1, q_2, A, B) \xrightarrow{x} (q_1, q_2', A \cup \mathrm{alph}(x), B')$$

  with $B' = B \cup \{2\}$ if $t_2 \in \Delta_2$ and $B' = B$ otherwise.

  Moreover, the projections of this transition are $(q_1, 1, q_1)$ and t_2 respectively.

- Finally, we add the final 1-transitions

$$(q_1, q_2, A, \{1,2\}) \xrightarrow{1} (q_1, q_2, A, \emptyset)$$

whose projections are $(q_1, 1, q_1)$ and $(q_2, 1, q_2)$ respectively.

Let us show that the language accepted by this automaton is $K' \sqcup\!\sqcup_I L'$. Let P be an accepted path of $\mathcal{A}$ with label w. Clearly, the projections P_1 and P_2 of P are paths of $\mathcal{A}_1$ and $\mathcal{A}_2$. Let u and v be the labels of P_1 and P_2. From the definition of the transitions, we can easily verify that $w \in u \sqcup\!\sqcup_I v$. Moreover, between two final transitions of P there must be at least one final transition in the corresponding parts of P_1 and P_2. Therefore, P_1 and P_2 are accepted paths and $w \in K' \sqcup\!\sqcup_I L'$.

Conversely, let $w \in K' \sqcup\!\sqcup_I L'$. We have $w = u_1 v_1 u_2 v_2 \ldots$ with $u = u_1 u_2 \ldots \in K'$, $v = v_1 v_2 \ldots \in L'$ and $(v_i, u_j) \in I$ for all $i < j$. Let

$$q_{1,0} = p_0 \xrightarrow{*}{}^{u_1} p_1 \xrightarrow{*}{}^{u_2} p_2 \xrightarrow{*}{}^{u_3} p_3 \ldots$$

be an accepting path in $\mathcal{A}_1$ for u and let

$$q_{2,0} = q_0 \xrightarrow{*}{}^{v_1} q_1 \xrightarrow{*}{}^{v_2} q_2 \xrightarrow{*}{}^{v_3} q_3 \ldots$$

be an accepting path in $\mathcal{A}_2$ for v. We construct an accepting path for w in $\mathcal{A}$ as follows:

$$\begin{array}{llllll}
 & (p_0, q_0, \emptyset, \emptyset) & \xrightarrow{*}{}^{u_1} & (p_1, q_0, \emptyset, B_1) & \xrightarrow{*}{}^{v_1} & (p_1, q_1, \mathrm{alph}(v_1), B_1') \\
\xrightarrow{1} & (p_1, q_1, \mathrm{alph}(v_1), B_1'') & \xrightarrow{*}{}^{u_2} & (p_2, q_1, \mathrm{alph}(v_1), B_2) & \xrightarrow{*}{}^{v_2} & (p_2, q_2, \mathrm{alph}(v_1 v_2), B_2') \\
\xrightarrow{1} & (p_2, q_2, \mathrm{alph}(v_1 v_2), B_2'') & \xrightarrow{*}{}^{u_3} & (p_3, q_2, \mathrm{alph}(v_1 v_2), B_3) & \xrightarrow{*}{}^{v_3} & \ldots
\end{array}$$

with for all $i \geq 1$, $B_i'' = \emptyset$ if $B_i' = \{1,2\}$ and $B_i'' = B_i'$ otherwise. $\square$

Proposition 2.10 ((Gastin, 1991)) *Let $K, L \subseteq \mathbb{R}(X, D)$ be recognizable (rational respectively) real trace languages such that for all $x \in K$, $y \in L$ it holds* $\mathrm{alphinf}(x) \times \mathrm{alph}(y) \subseteq I$. *Then $K \cdot L$ is recognizable (rational respectively).*

Proof: Consider the canonical mapping $\varphi : X^\infty \to \mathbb{R}(X, D)$. First, observe that for $u, v \in X^\infty$, $u \sqcup\!\sqcup_I v$ is not empty if and only if $\mathrm{alphinf}(u) \times \mathrm{alph}(v) \subseteq I$. Moreover, in this case, $\varphi(w) = \varphi(u) \cdot \varphi(v)$ for all $w \in u \sqcup\!\sqcup_I v$. Now, there exist recognizable word languages $K', L' \subseteq X^\infty$ such that $\varphi(K') = K$ and $\varphi(L') = L$. By the observation above, we have $\varphi(K' \sqcup\!\sqcup_I L') = K \cdot L$. Hence, by Lemma 2.9 and Proposition 2.1, $K \cdot L$ is rational.

Finally, if K and L are recognizable, we may assume that $K' = \varphi^{-1}(K)$ and $L' = \varphi^{-1}(L)$ are closed (Proposition 2.3). We claim that in this case, $K' \sqcup\!\sqcup_I L'$ is closed too, i.e., $K' \sqcup\!\sqcup_I L' = \varphi^{-1}(K \cdot L)$, which proves that $K \cdot L$ is recognizable.

Let $w \in \varphi^{-1}\varphi(K' \sqcup\!\sqcup_I L') = \varphi^{-1}(K \cdot L)$ then $\varphi(w) = x \cdot y$ for some $x \in K$ and $y \in L$. We denote by w_i the i-th letter of w or 1 if $|w| < i$. For a letter a and a word (or a real trace) x we denote by $|x|_a$ the number of occurrences of a in x. We construct two sequences $u_1, v_1, u_2, v_2, \ldots \in X \cup \{1\}$ as follows: $u_i = w_i$, $v_i = 1$ if $|x|_{w_i} \geq |w_1 \ldots w_i|_{w_i}$ and $u_i = 1$, $v_i = w_i$ otherwise.

Clearly, $w = u_1v_1u_2v_2\ldots$. Let $u = u_1u_2\ldots$ and $v = v_1v_2\ldots$. From $\varphi(w) = x \cdot y$ we deduce that $\varphi(u) = x$, $\varphi(v) = y$ and $(v_i, u_j) \in I$ for all $i < j$. Hence, $u \in \varphi^{-1}(K) = K'$, $v \in \varphi^{-1}(L) = L'$ and $w \in K' \sqcup\!\sqcup_I L'$ which proves the claim. □

Remark that Proposition 2.10 states that the family $\mathrm{Rat}(\mathbb{R}(X, D))$ is closed under this extended concatenation. Hence, we may have replaced the concatenation restricted to languages of finite traces on the left by this extended one without changing the family $\mathrm{Rat}(\mathbb{R}(X, D))$.

As stated in Proposition 2.1, a real trace language L is rational if and only if it can be written as a finite union of sets of the form $M \cdot N^\omega$ where $M, N \subseteq \mathbb{M}(X, D)$ are rational. For recognizable languages this can be strengthened to the following normal form result.

Proposition 2.11 ((Gastin et al., 1991)) *A real trace language $L \subseteq \mathbb{R}(X, D)$ is recognizable if and only if it can be written as a finite union of sets of the form $M \cdot N_1^\omega \cdots N_k^\omega = M \cdot (N_1 \cdots N_k)^\omega$ where $M, N_1, \ldots, N_k \subseteq \mathbb{M}(X, D)$ are recognizable languages such that all traces in N_i are connected and have the same alphabet A_i with $A_i \times A_j \subseteq I$ for $i \neq j$.*

Proof: First, it is clear by Remark 2.2, Corollary 2.8 and Proposition 2.10 that every such finite union is recognizable.

The original proof (Gastin et al., 1991) for the other direction uses Mezei's Theorem. Here we simply use Ramsey factorization.

Let $\eta' : \mathbb{M}(X, D) \to S'$ be a morphism recognizing $L \subseteq \mathbb{R}(X, D)$. We define the direct product $S = S' \times \mathcal{P}(X)$ where the operation on the power set $\mathcal{P}(X)$ is union and the morphism $\eta : \mathbb{M}(X, D) \to S$ by $\eta(x) = (\eta'(x), \mathrm{alph}(x))$ for all $x \in \mathbb{M}(X, D)$. Note that η recognizes L, too. Moreover, for $s \in S$, the second projection $\mathrm{alph}(s)$ is well-defined and satisfies $\mathrm{alph}(x) = \mathrm{alph}(\eta(x))$ for all $x \in \mathbb{M}(X, D)$. This property will be used in the remainder of the proof.

Using Ramsey factorization, recall that we obtain (Gastin, 1991):

$$L = \bigcup\nolimits_{(s,e)\in P} \eta^{-1}(s) \cdot \eta^{-1}(e)^\omega$$

where $P = \{(s, e) \in S \times S \mid se = s, e^2 = e \text{ and } \eta^{-1}(s) \cdot \eta^{-1}(e)^\omega \cap L \neq \emptyset\}$. For each $(s, e) \in P$, we decompose the alphabet $A = \mathrm{alph}(e)$ into connected components, $A = \bigcup_{i=1}^k A_i$ such that A_i is not empty and connected and $A_i \times A_j \subseteq I$ for all $i \neq j$. Moreover, we define the finite set

$$E = \{(e_1, \ldots, e_k) \mid e = e_1 \cdots e_k \text{ and } \mathrm{alph}(e_i) = A_i, \text{ for all } 1 \leq i \leq k\}$$

We claim that

$$\eta^{-1}(s)\eta^{-1}(e)^\omega = \bigcup\nolimits_{(e_1,\ldots,e_k)\in E} \eta^{-1}(s)\eta^{-1}(e_1)^\omega \cdots \eta^{-1}(e_k)^\omega$$

which concludes the proof. Note that, in the definition of E, we could require in addition that $se_i = s$ and $e_i^2 = e_i$ but this is not necessary for the proof of this proposition.

From the alphabetic property of η, we obtain

$$\eta^{-1}(e_1)^\omega \cdots \eta^{-1}(e_k)^\omega = (\eta^{-1}(e_1) \cdots \eta^{-1}(e_k))^\omega$$

Moreover, $\eta^{-1}(e_1)\cdots\eta^{-1}(e_k) \subseteq \eta^{-1}(e)$, hence one inclusion is clear.

Conversely, we define a morphism $\psi : \eta^{-1}(e) \to E$ as follows. Let y be in $\eta^{-1}(e)$. Since $\mathrm{alph}(y) = \mathrm{alph}(e) = A$, y admits a unique decomposition in connected components $y = y_1 \cdots y_k$ such that $\mathrm{alph}(y_i) = A_i$ for all $1 \le i \le k$. Then we set $\psi(y) = (\eta(y_1), \ldots, \eta(y_k))$.

Now, let $x_0x_1x_2\ldots \in \eta^{-1}(s)\eta^{-1}(e)^\omega$ with $x_0 \in \eta^{-1}(s)$ and $x_i \in \eta^{-1}(e)$ for all $i \ge 1$. Using Ramsey factorization, we obtain an infinite sequence of integers $0 < i_1 < i_2 < i_3 < \ldots$ and a tuple $(e_1, \ldots, e_k) \in E$ such that $\psi(x_{1+i_j}\ldots x_{i_k}) = (e_1, \ldots, e_k)$ for all $1 \le j < k$. Hence, $x_{1+i_j}\cdots x_{i_{j+1}} \in \eta^{-1}(e_1)\cdots\eta^{-1}(e_k)$ for all $j \ge 1$ and we obtain $x_{1+i_1}x_{2+i_1}x_{3+i_1}\ldots \in (\eta^{-1}(e_1)\cdots\eta^{-1}(e_k))^\omega = \eta^{-1}(e_1)^\omega\cdots\eta^{-1}(e_k)^\omega$. This proves the claim since $\eta(x_0x_1\ldots x_{i_1}) = se\cdots e = se = s$. $\square$

Corollary 2.12 *The family of recognizable real languages is the smallest family which contains finite sets of finite traces and which is closed under union, concatenation restricted to languages of finite traces on the left, and Kleene $*$ and ω-iteration both restricted to connected languages of finite traces.*

The final results of this section show the closure of rational and recognizable languages by left and right quotient. These results will become crucial in the next section.

Proposition 2.13 *Let $K \subseteq \mathbb{R}(X,D)$ be arbitrary and $L \subseteq \mathbb{R}(X,D)$ be recognizable. Then the following left and right quotients are recognizable, too:*

$$\begin{aligned} K^{-1}\cdot L &= \{y \in \mathbb{R}(X,D) \mid xy \in L \text{ for some } x \in K \text{ with } \mathrm{alphinf}(x) \times \mathrm{alph}(y) \subseteq I\} \\ L\cdot K^{-1} &= \{x \in \mathbb{R}(X,D) \mid xy \in L \text{ for some } y \in K \text{ with } \mathrm{alphinf}(x) \times \mathrm{alph}(y) \subseteq I\} \end{aligned}$$

If, in addition, $K \subseteq \mathbb{R}(X,D)$ is recognizable, then these quotients are effectively computable.

Proof: Let $\eta : \mathbb{M}(X,D) \longrightarrow S$ be a morphism onto a finite monoid S which recognizes L and let $X_s = \eta^{-1}(s)$ for $s \in S$. As in the proof of Proposition 2.11, we may also assume that $\mathrm{alph}(s)$ is well-defined for $s \in S$ and $\mathrm{alph}(x) = \mathrm{alph}(s)$ for all $x \in X_s$. For $s, t \in S$, we write $(s,t) \in I$ if $\mathrm{alph}(s) \times \mathrm{alph}(t) \subseteq I$.

We claim that η recognizes the language $K^{-1}\cdot L$. Let $(y_i)_{i<\omega}$ be any sequence of finite traces such that $y = \prod_{i<\omega} y_i \in K^{-1}\cdot L$. By definition, there exists some real trace $x \in K$ such that $x \cdot y \in L$ and $\mathrm{alphinf}(x) \times \mathrm{alph}(y) \subseteq I$. Let $(x_i)_{i<\omega}$ be a sequence of finite traces such that $x = \prod_{i<\omega} x_i$ and $\mathrm{alph}(x_i) = \mathrm{alphinf}(x)$ for all $i \ge 1$. It follows $x \cdot y = (x_0y_0)(x_1y_1)\ldots \in L$.

Now, let $(z_i)_{i<\omega}$ be a sequence of finite traces such that $\eta(z_i) = \eta(y_i)$ for all $i < \omega$ and let $z = \prod_{i<\omega} z_i$. Since η recognizes L, we get $(x_0z_0)(x_1z_1)\ldots \in L$. But from the alphabetic property of η, $(x_i, z) \in I$ for all $i \ge 1$ and thus $x \cdot z = (x_0z_0)(x_1z_1)\ldots \in L$. Therefore $z \in K^{-1}\cdot L$ and the claim is proved.

Let R be any real trace language recognized by the morphism η. Recall that using Ramsey factorization we obtain (Gastin, 1991):

$$R = \bigcup\nolimits_{(s,e)\in P(R)} \eta^{-1}(s)\cdot\eta^{-1}(e)^\omega$$

where $P(R) = \{(s,e) \in S \times S \mid se = s, e^2 = e \text{ and } \eta^{-1}(s) \cdot \eta^{-1}(e)^\omega \cap R \neq \emptyset\}$.

Suppose now that K is recognizable. We may assume that the morphism η recognizes both languages K and L. It is then easy to verify that $P(K^{-1} \cdot L) = \{(s,e) \in S \times S \mid se = s, e^2 = e, \exists (r,d) \in P(K) \text{ such that } (d, se) \in I, (rs, de) \in P(L)\}$. Hence the quotient $K^{-1} \cdot L$ is effectively computable.

The proof for $L \cdot K^{-1}$ is symmetric. $\square$

For rational languages such a general result does not hold. We have the following counter example.

Example: Let $\mathbb{M}(X,D) = \{a,b\}^* \times \{c,d\}^* \times \{e\}^*$ be a direct product of three free monoids. Consider the following two rational sets: $K = (ae)^*b$ and $L = (ac)^*b(de)^*$. Then we have $K^{-1}L \subseteq \{c,d\}^* \times \{e\}^*$. The projection of $K^{-1}L$ to $\{c,d\}^*$ yields the non-rational set $\{c^n d^m \mid n \leq m\}$. In order to have an example of ω- languages, simply substitute b by b^ω.

However, the quotient of a rational language by a recognizable language is effectively rational. This is stated in the next Proposition.

Proposition 2.14 *Let $K \subseteq \mathbb{R}(X,D)$ be recognizable and $L \subseteq \mathbb{R}(X,D)$ be rational. Then the following left and right quotients are effectively computable and rational, too:*

$$\begin{aligned} K^{-1} \cdot L &= \{y \in \mathbb{R}(X,D) \mid xy \in L \text{ for some } x \in K \text{ with } \operatorname{alphinf}(x) \times \operatorname{alph}(y) \subseteq I\} \\ L \cdot K^{-1} &= \{x \in \mathbb{R}(X,D) \mid xy \in L \text{ for some } y \in K \text{ with } \operatorname{alphinf}(x) \times \operatorname{alph}(y) \subseteq I\} \end{aligned}$$

Proof: We give the proof for $K^{-1} \cdot L$ only. The other case $L \cdot K^{-1}$ is obtained analogously. Let $K' = \varphi^{-1}(K)$ and $L' \subseteq X^\infty$ be a recognizable language such that $\varphi(L') = L$. Let X_1, X_2 be two copies of X and let $\hat{X} = X_1 \cup X_2$. Define three morphisms h, p_1, p_2 from $\hat{X}^\infty$ in X^∞ by $h(a_i) = a$, $p_i(a_i) = a$ and $p_j(a_i) = 1$ for $i,j = 1,2$ with $i \neq j$. Finally, define the recognizable language $W \subseteq \hat{X}^\infty$ by its complement $\overline{W} = \cup_{(a,b) \in D} \hat{X}^* b_2 \hat{X}^* a_1 \hat{X}^\infty$.

Since recognizable word languages are closed under morphism, inverse morphism and intersection, the language $N' = p_2(p_1^{-1}(K') \cap h^{-1}(L') \cap W)$ is recognizable, too. Moreover, it is easy to verify that

$$\begin{aligned} N' &= \{v_1 v_2 \ldots \in X^\infty \mid \exists u_1 u_2 \ldots \in K' \text{ with } u_1 v_1 u_2 v_2 \ldots \in L' \\ &\qquad \text{and } (v_i, u_j) \in I \text{ for all } i < j\} \\ &= \{v \in X^\infty \mid (u \sqcup\!\sqcup_I v) \cap L' \neq \emptyset \text{ for some } u \in K'\} \end{aligned}$$

We claim that $\varphi(N') = K^{-1} \cdot L$ which proves that $K^{-1} \cdot L$ is rational.

Let v be in N'. There exists $w \in (u \sqcup\!\sqcup_I v) \cap L'$ for some $u \in K'$. In the proof of Proposition 2.10 we have shown that $\varphi(w) = \varphi(u) \cdot \varphi(v)$. Therefore, $\varphi(v) \in K^{-1} \cdot L$.

Conversely, let y be in $K^{-1} \cdot L$. There exists $x \in K$ such that $\operatorname{alphinf}(x) \times \operatorname{alph}(y) \subseteq I$ and $z = xy \in L$. Let $w \in \varphi^{-1}(z) \cap L'$. In the proof of Proposition 2.10 we have shown that $\varphi^{-1}(xy) = \varphi^{-1}(x) \sqcup\!\sqcup_I \varphi^{-1}(y)$. Hence, there exist $u \in \varphi^{-1}(x)$ and $v \in \varphi^{-1}(y)$ such that $w \in u \sqcup\!\sqcup_I v$, whence $(u \sqcup\!\sqcup_I v) \cap L' \neq \emptyset$. Since $u \in \varphi^{-1}(x) \subseteq \varphi^{-1}(K) = K'$, we obtain $v \in N'$. $\square$

3 Rational and recognizable complex trace languages

We begin, in this section, the study of rational and recognizable languages of complex traces. We use a definition which says that a complex trace language is recognizable (rational respectively) if certain real parts have the same property. This allows a uniform treatment for rational and recognizable languages. Moreover, in this way, results are more easily infered from real trace languages to complex trace languages. Finally, these definitions will be justified by various characterizations. For instance, the family of recognizable complex trace languages can be characterized using recognizing morphisms with respect to $(\omega+1)$-sequences of finite traces (Proposition 3.11) or by recognizing morphisms with respect to arbitrary ordinal sequences, see Section 3.4.

The family of rational complex trace languages is, as usual, the smallest family of complex trace languages containing the finite languages of finite traces and closed under union, concatenation, Kleene-$*$ and ω-iteration (Corollary 3.8).

Closure properties under union, intersection, complementation, concatenation and iterations are studied for the families of rational and of recognizable complex trace languages. Then, normal forms for rational and recognizable languages of complex traces are proposed in Corollary 3.6 extending the corresponding results on real trace languages.

Finally, the family of recognizable (rational respectively) complex trace languages is shown to be closed under quotients (quotients by recognizable sets respectively) in Theorem 3.10.

3.1 Definitions

We first define rational and recognizable complex trace languages. There will be various equivalent characterizations for these notions which justify the following definition. Recall that if $L \subseteq \mathbb{C}(X,D)$ is a complex trace language and $A \subseteq X$ is a subset of the alphabet, we denote by $L_A = \{x \in L \mid \mathrm{Im}(x) = D(A)\}$.

Definition: *A complex trace language $L \subseteq \mathbb{C}(X,D)$ is called rational (recognizable respectively) if for all $A \subseteq X$ the real language $\mathrm{Re}(L_A)$ is rational (recognizable respectively). The families are denoted by $\mathrm{Rat}(\mathbb{C}(X,D))$ and $\mathrm{Rec}(\mathbb{C}(X,D))$ respectively.*

Proposition 3.1 *The family $\mathrm{Rec}(\mathbb{C}(X,D))$ is a boolean algebra: it is closed under union, intersection, and complementation. The family $\mathrm{Rat}(\mathbb{C}(X,D))$ is closed under union. Moreover, the intersection of a rational complex trace language by a recognizable complex trace language is rational.*

Proof: Since $\mathrm{Re} : \mathbb{C}(X,D) \to \mathbb{R}(X,D)$ commutes with union, it follows from results of Section 2 on real traces that $\mathrm{Rat}(\mathbb{C}(X,D))$ and $\mathrm{Rec}(\mathbb{C}(X,D))$ are closed under union. The closure of $\mathrm{Rec}(\mathbb{C}(X,D))$ under intersection and complementation follows from Remark 2.2 and the formulae: for all $K, L \subseteq \mathbb{C}(X,D)$ and $A \subseteq X$, $\mathrm{Re}(K_A \cap L_A) = \mathrm{Re}(K_A) \cap \mathrm{Re}(L_A)$ and $\mathrm{Re}((\mathbb{C}(X,D)\setminus K)_A) = \mathrm{Re}(\mathbb{C}(X,D)_A)\setminus\mathrm{Re}(K_A)$. Finally, the last result follows directly from Lemma 2.4. $\square$

For any real trace language $L \subseteq \mathbb{R}(X,D)$ and any subset $A \subseteq X$, it holds $\mathrm{Re}(L_A) = \{r \in L \mid \mathrm{Im}(r)\} = D(A)\} = L \cap \mathbb{R}(X,D)_A$. Since

$$\varphi^{-1}(\mathbb{R}(X,D)_A) = \bigcup_{\{B \subseteq X \mid D(B)=D(A)\}} \{x \in X^\infty \mid \mathrm{alphinf}(x) = B\}$$

the set $\mathbb{R}(X,D)_A$ is a recognizable real trace language. Therefore if L is a rational (recognizable respectively) real trace language, then so is $Re(L_A)$ for any $A \subseteq X$. Hence L is also a rational (recognizable respectively) complex trace language. More precisely, we have (as expected) $\mathrm{Rat}(\mathbb{R}(X,D)) = \{L \subseteq \mathbb{R}(X,D) \mid L \in \mathrm{Rat}(\mathbb{C}(X,D))\}$ and $\mathrm{Rec}(\mathbb{R}(X,D)) = \{L \subseteq \mathbb{R}(X,D) \mid L \in \mathrm{Rec}(\mathbb{C}(X,D))\}$. At last, note that since $\mathrm{Re}(L) = \bigcup_{A \subseteq X} \mathrm{Re}(L_A)$, we have also the inclusions $\mathrm{Re}(\mathrm{Rat}(\mathbb{C}(X,D))) \subseteq \mathrm{Rat}(\mathbb{R}(X,D))$ and $\mathrm{Re}(\mathrm{Rec}(\mathbb{C}(X,D))) \subseteq \mathrm{Rec}(\mathbb{R}(X,D))$. The situation is summarized in the following figure.

$$\begin{array}{ccccc}
\mathrm{Rec}(\mathbb{R}(X,D)) & \subseteq & \mathrm{Rec}(\mathbb{C}(X,D)) & \xrightarrow{\mathrm{Re}} & \mathrm{Rec}(\mathbb{R}(X,D)) \\
\cap| & & \cap| & & \cap| \\
\mathrm{Rat}(\mathbb{R}(X,D)) & \subseteq & \mathrm{Rat}(\mathbb{C}(X,D)) & \xrightarrow{\mathrm{Re}} & \mathrm{Rat}(\mathbb{R}(X,D))
\end{array}$$

Before we continue the reader should be convinced that it makes no sense to define a language $L \subseteq \mathbb{C}(X,D)$ to be rational (recognizable respectively) if the projection $\mathrm{Re}(L) \subseteq \mathbb{R}(X,D)$ is rational (recognizable respectively). Both classes of languages would be uncountable. (Note that the families of rational and recognizable complex trace languages are countable. Indeed, a complex trace language is entirely defined by the languages L_A for $A \subseteq X$ and the family of rational real trace languages is countable since it is defined by rational expressions.) For instance, if $(X,D) = a — b — c$, for any $P \subseteq \mathbb{N}$, the language $L_P = \{(\varphi(b)^j\varphi(a)^\omega, D(a)) \mid j \in P\} \cup \{(\varphi(b)^j\varphi(a)^\omega, D(a,b)) \mid j \notin P\}$ verifies $\varphi^{-1}(\mathrm{Re}(L_P)) = a^*b^\omega$ and would be therefore recognizable. Note also that the family of recognizable languages would not be closed under intersection. Namely if $(X,D) = a — b — c — d$ and $L = \varphi(ac)^*\varphi(b)^\omega\varphi(c) \cup \{\varphi(a),\varphi(c)\}^*\varphi(b)^\omega$, $\mathrm{Re}(L) = \{\varphi(a),\varphi(c)\}^*\varphi(b)^\omega$ is a recognizable real trace language whereas $L_{\{b,c\}} = L \cap \mathbb{C}(X,D)_{\{b,c\}}$ is not recognizable since $\mathrm{Re}(L_{\{b,c\}}) = \varphi(ac)^*\varphi(b)^\omega$. Similarly, since the canonical mapping $\varphi : X^\infty \to \mathbb{C}(X,D)$ is not surjective, it cannot be used to define a language $L \subseteq \mathbb{C}(X,D)$ to be recognizable. For example, simply requiring that $\varphi^{-1}(L) \subseteq X^\infty$ has to be recognizable would yield that any language not containing real traces is recognizable. (Of course, according to our definition, $L \in Rec(\mathbb{C}(X,D))$ implies that $\varphi^{-1}(L) \subseteq X^\infty$ is recognizable.)

3.2 Technical results

We prove in this section some technical results that will be used to prove closure properties of rational and recognizable complex trace languages.

Recall the following notations. Let $L \subseteq \mathbb{C}(X,D)$ be a complex trace language and $A \subseteq X$ be a subset of the alphabet. We denote by $\mu_A(L) = \{\mu_A(x) \mid x \in L\}$ where $\mu_A(x)$ is the maximal real prefix of $\mathrm{Re}(x)$ containing letters independent of A, only. It will be useful to have a finer partition. Let $L \subseteq \mathbb{C}(X,D)$ and $A, B \subseteq X$. We define $L_{A,B} = \{x \in L \mid \mathrm{Im}(yx) = D(B) \text{ for some } y \in \mathbb{C}(X,D)_A\}$. Note that in the definition above the existential "for some" can be replaced by "for all". The reason to introduce $L_{A,B}$ is that if we concatenate any complex trace $y \in \mathbb{C}(X,D)_A$ with some $x \in L$ such that $yx \in \mathbb{C}(X,D)_B$, then in fact $x \in L_{A,B}$. Note that in this case $\mathrm{Re}(yx) = \mathrm{Re}(y)\mu_A(x)$, therefore, we are also interested in the language $\mu_{A,B}(L)$ which is defined by $\mu_{A,B}(L) = \mu_A(L_{A,B})$.

The formulae given in the next lemma will be useful throughout.

Lemma 3.2 *Let $K, L \subseteq \mathbb{C}(X,D)$ be complex trace languages and $B \subseteq X$. Then the following formulae hold:*

i) $L_B = L_{\emptyset,B}$

ii) $L = \bigcup_{C\subseteq X} L_{B,C}$

iii) $\mu_B(L) = \bigcup_{C\subseteq X} \mu_{B,C}(L)$

iv) $(KL)_B = \bigcup_{A\subseteq X} K_A L_{A,B}$

v) $\mathrm{Re}((KL)_B) = \bigcup_{A\subseteq X} \mathrm{Re}(K_A)\mu_{A,B}(L)$

Proof: i) is obvious. ii) Let $x \in L$ and $y \in \mathbb{C}(X,D)_B$. Then, $D(B) \subseteq \mathrm{Im}(yx) = D(C)$ for some $C \subseteq X$. Therefore $x \in L_{B,C}$. iii) follows since μ_A commutes with union. iv) We have $(KL)_B = \bigcup_{A\subseteq X}(K_A L)_B = \bigcup_{A\subseteq X}(K_A L_{A,B})$. At last, for any $x \in K_A$ and any $y \in L_{A,B}$, $\mathrm{Re}(xy) = \mathrm{Re}(x)\mu_A(y)$ and v) follows from iv). $\square$

Finally, we define $\nu_A(L) = \mu_{A,A}(L)$. Each trace in $\nu_A(L)$ is finite since $\mathrm{alph}(u) \subseteq I(A)$ and $\mathrm{Im}(u) \subseteq D(A)$ implies $\mathrm{Im}(u) = \emptyset$. Note also that $\nu_\emptyset(L)$ is just the finitary part $L_\emptyset = L \cap \mathbb{M}(X,D)$. Later, we will introduce finite components, which could be characterized as the union of $\nu_A(L)$ over $A \subseteq X$. For the moment we are interested in the $\nu_A(L)$, because we have the following formula.

Lemma 3.3 *Let $L \subseteq \mathbb{C}(X,D)$ be a complex trace language and $\dagger \in \{*, \omega\}$. Let $\mathcal{A}$ be the finite set of sequences $\mathcal{A} = \{(A_0,\dots,A_k) \mid \emptyset = D(A_0) \subsetneq \cdots \subsetneq D(A_k) \subseteq X\}$. Then the language $L^\dagger$ is the finite union*

$$L^\dagger = \bigcup\nolimits_{(A_0,\dots,A_k)\in\mathcal{A}} \nu_{A_0}(L)^* L_{A_0,A_1} \dots \nu_{A_{k-1}}(L)^* L_{A_{k-1},A_k} \nu_{A_k}(L)^\dagger$$

Proof: The proof uses in a crucial way the observation that, if $x, y \in \mathbb{C}(X,D)$ and $Im(x) = Im(xy) = D(A)$ then $\mu_A(y) = \nu_A(y)$ and $xy = x\nu_A(y)$.

Let $x = x_1 x_2 \ldots \in L^\dagger$, $x_i \in L$. Eventually, the sequence of alphabets $\mathrm{Im}(x_1) \subseteq \mathrm{Im}(x_1 x_2) \subseteq \cdots$ becomes stationary. Therefore, there exist an integer k, a sequence of integers $0 = i_0 < i_1 < \ldots < i_k < i_{k+1} = \omega$ and a sequence $(A_0,\dots,A_k) \in \mathcal{A}$ such that $Im(x_1 \ldots x_n) = D(A_j)$ for all $i_j \le n < i_{j+1}$ and $0 \le j \le k$. Define now the sequence $(y_n)_{n\ge1}$ by $y_n = \nu_{A_j}(x_n) \in \nu_{A_j}(L)$ if $i_j < n < i_{j+1}$ for some $0 \le j \le k$ and by $y_n = x_n \in L_{A_{j-1},A_j}$ if $n = i_j$ for some $1 \le j \le k$. From the observation above, it follows $x_1 \ldots x_n = y_1 \ldots y_n$ for all n and thus $x = y_1 y_2 \ldots y_n \ldots$. Therefore,

$$x \in \nu_{A_0}(L)^* L_{A_0,A_1} \dots \nu_{A_{k-1}}(L)^* L_{A_{k-1},A_k} \nu_{A_k}(L)^\dagger$$

Conversely, we show by induction on k that $\nu_{A_0}(L)^* L_{A_0,A_1} \dots \nu_{A_{k-1}}(L)^* L_{A_{k-1},A_k} \nu_{A_k}(L)^\dagger \subseteq L^\dagger$, for any $(A_0, \dots, A_k) \in \mathcal{A}$. For $k = 0$ this formula becomes $\nu_\emptyset(L)^\dagger = L_\emptyset^\dagger \subseteq L^\dagger$ which is clear. Assume that, for any $(A_0, \dots, A_k) \in \mathcal{A}$, the inclusion holds and let $(A_0, \dots, A_{k+1}) \in \mathcal{A}$. By induction hypothesis applied to $(A_0, \dots, A_k)$, we have in particular:

$$\nu_{A_0}(L)^* L_{A_0,A_1} \dots \nu_{A_{k-1}}(L)^* L_{A_{k-1},A_k} \nu_{A_k}(L)^* \subseteq L^* \tag{4}$$

Since all the complex traces belonging to the left hand side of Equation (4) have clearly $D(A_k)$ as imaginary part, it follows:

$$\nu_{A_0}(L)^* L_{A_0,A_1} \dots \nu_{A_{k-1}}(L)^* L_{A_{k-1},A_k} \nu_{A_k}(L)^* \subseteq (L^*)_{A_k}$$

Therefore the inclusion for $(A_0, \dots, A_{k+1})$ follows since $(L^*)_{A_k} L_{A_k,A_{k+1}} \subseteq (L^*)_{A_{k+1}}$ and, from the observation above, $(L^*)_{A_{k+1}} \nu_{A_{k+1}}(L)^\dagger \subseteq L^\dagger$ for $\dagger \in \{*, \omega\}$. $\square$

Lemma 3.4 *Let $L \subseteq \mathbb{C}(X,D)$ be rational (recognizable respectively) and $A, B \subseteq X$. Then the languages $L_{A,B} \subseteq \mathbb{C}(X,D)$ and $\mu_{A,B}(L) \subseteq \mathbb{R}(X,D)$ are rational (recognizable respectively).*

Proof: First, we show the assertion for $L_{A,B}$. Note that $(L_{A,B})_C = (L_C)_{A,B}$ for all $C \subseteq X$. Thus,

$$\begin{aligned}
\mathrm{Re}((L_{A,B})_C) &= \mathrm{Re}((L_C)_{A,B}) \\
&= \mathrm{Re}(\{x \in L_C \mid \mathrm{Im}(yx) = D(B) \text{ for some } y \in \mathbb{C}(X,D)_A\}) \\
&= \{r \in \mathrm{Re}(L_C) \mid \mathrm{Im}(yr) \cup D(C) = D(B) \text{ for some } y \in \mathbb{C}(X,D)_A\} \\
&= \mathrm{Re}(L_C) \cap K
\end{aligned}$$

where $K = \{r \in \mathbb{R}(X,D) \mid D(A \cup C \cup \mathrm{Suff}_A(r)) = D(B)\}$.

Using Remark 2.2 and Lemma 2.4, it is sufficient to prove that the language K is recognizable. Indeed, it is easy to verify that the language $\varphi^{-1}(K)$ is accepted by the following deterministic Büchi automaton. The set of states is $\mathcal{P}(X)$, the initial state is A and for all $q \in \mathcal{P}(X)$ and $a \in X$ we have the transition (q, a, q') where $q' = q \cup \{a\}$ if $a \in D(q)$ and $q' = q$ otherwise. A transition (q, a, q') is final if and only if $D(A \cup C \cup q) = D(B)$.

For $\mu_{A,B}(L)$ we have $\mu_{A,B}(L) = \mu_A(L_{A,B})$ by definition. Hence it is enough to show the assertion for $\mu_A(L)$. Since $\mu_A(L) = \mu_A(\mathrm{Re}(L))$, we are reduced to consider real trace language $L \subseteq \mathbb{R}(X,D)$.

For $L \subseteq \mathbb{R}(X,D)$ we have $\mu_A(L) = M \cap (L \cdot \mathrm{Suff}_A^{-1})$ where $M = \{x \in \mathbb{R}(X,D) \mid \mathrm{alph}(x) \subseteq I(A)\}$ and

$$\mathrm{Suff}_A = \{r \in \mathbb{R}(X,D) \mid \min(r) \subseteq D(A)\} = \mathbb{R}(X,D) \setminus \bigcup_{a \in I(A)} a \cdot \mathbb{R}(X,D)$$

Since M and Suff_A are recognizable, the results follow from Lemma 2.4, Propositions 2.13, and 2.14. $\square$

3.3 Closure properties

So far, we have allowed as rational operation on real traces the concatenation $K \cdot L$, only if $K \subseteq \mathbb{M}(X, D)$. This was motivated by the fact that $\mathbb{R}(X, D)$ is not a monoid. For complex traces such a restriction is obviously not adequate and our first result underlines this. It follows directly from Lemma 3.2, v), Proposition 2.10 and Lemma 3.4.

Theorem 3.5 *Let $K, L \in \mathbb{C}(X, D)$ be rational (recognizable respectively). Then the concatenation $K \cdot L$ has the same property.*

We can now extend Proposition 2.1 and Proposition 2.11 and give normal forms for rational and recognizable complex trace languages.

Corollary 3.6 *i) A language $L \subseteq \mathbb{C}(X, D)$ is rational if and only if it can be written as a finite union of sets of the form $MN^\omega s$ where $M, N \subseteq \mathbb{M}(X, D)$ are rational and $s \in \mathbb{M}(X, D)$ is a finite (shift-)trace.*

ii) A language $L \subseteq \mathbb{C}(X, D)$ is recognizable if and only if it can be written as a finite union of sets of the form $MN_1^\omega \dots N_k^\omega s$ where $M, N_1, \dots, N_k \subseteq \mathbb{M}(X, D)$ are recognizable, each trace in N_i has the same connected alphabet A_i, $A_i \times A_j \subseteq I$ for $i \neq j$, and $s \in \mathbb{M}(X, D)$ is a finite (shift-)trace.

Proof: We have seen in the preliminaries every complex trace language $L \subseteq \mathbb{C}(X, D)$ can be written as a finite union:

$$L = \bigcup_{A \text{ shift } B} (\mathrm{Re}(L_B) \cap \mathbb{R}(X, D)_A) s_{A,B}$$

where $s_{A,B}$ is some finite shift trace from $D(A)$ to $D(B)$. Hence if L is rational (recognizable respectively), Proposition 2.1 (Proposition 2.11 respectively) applied to languages $\mathrm{Re}(L_B) \cap \mathbb{R}(X, D)_A$ yield the only-if parts of the corollary. The if-parts come from the inclusions $\mathrm{Rat}(\mathbb{R}(X, D)) \subseteq \mathrm{Rat}(\mathbb{C}(X, D))$ ($\mathrm{Rec}(\mathbb{R}(X, D)) \subseteq \mathrm{Rec}(\mathbb{C}(X, D))$ respectively) and the closures under concatenation, Theorem 3.5. □

Since the concatenation and the ω-product are fully defined in $\mathbb{C}(X, D)$, the $*$-iteration and the ω-iteration should be allowed as rational operations, in full generality. The next theorem states that this is indeed the case.

Theorem 3.7 *Let $L \in \mathrm{Rat}(\mathbb{C}(X, D))$ be a rational complex trace language. Then L^* and L^ω are rational.*

Proof: If L is a rational complex trace language, from Lemma 3.4, we deduce in particular that $\nu_A(L) \subseteq \mathbb{M}(X, D)$ is a rational language of finite traces for all $A \subseteq X$. Hence, for all $A \subseteq X$, $\nu_A(L)^*$ and $\nu_A(L)^\omega$ are rational real trace languages, too. Thus, the theorem follows from Lemma 3.3, Proposition 3.1 and Theorem 3.5. □

From closure under union, Theorems 3.5, 3.7 and Corollary 3.6, we obtain the following characterization which could also be used for a convenient definition of rational complex trace languages.

Corollary 3.8 *The family of rational complex trace languages is the smallest family which contains all rational languages of finite traces and which is closed under union, concatenation, Kleene-$*$, and ω-iteration.*

Of course, we cannot expect that recognizable languages are closed under Kleene-$*$ or ω-iteration, in general. However, let L be a recognizable complex trace language such that $\nu_A(L)^* \subseteq \mathbb{M}(X,D)$ is recognizable for all $A \subseteq X$. From Theorem 2.7, we deduce that, for all $A \subseteq X$, $\nu_A(L)^\omega$ is recognizable, too. Hence, from Lemma 3.3 and closures under union and concatenation we obtain the following corollary. It is our strongest result concerning the question when the star- and ω-iteration of a recognizable complex trace language is recognizable.

Corollary 3.9 *Let $L \subseteq \mathbb{C}(X,D)$ be a recognizable complex trace language such that $\nu_A(L)^* \subseteq \mathbb{M}(X,D)$ is recognizable for all $A \subseteq X$. Then L^* and L^ω are recognizable complex trace languages, too.*

We finish this section by an extension of Propositions 2.13 and 2.14 to complex trace languages.

Theorem 3.10 *Let $K \subseteq \mathbb{C}(X,D)$ be arbitrary (recognizable respectively) and $L \subseteq \mathbb{C}(X,D)$ be recognizable (rational respectively). Then the following left and right quotients are recognizable (rational respectively), too:*

$$\begin{aligned} L \cdot K^{-1} &= \{x \in \mathbb{C}(X,D) \mid xy \in L \text{ for some } y \in K\} \\ K^{-1} \cdot L &= \{y \in \mathbb{C}(X,D) \mid xy \in L \text{ for some } x \in K\} \end{aligned}$$

If K is recognizable, then the quotients are effectively computable.

Proof: First we consider $\mathrm{Re}((L \cdot K^{-1})_A)$ for some $A \subseteq X$. We have

$$(L \cdot K^{-1})_A = \bigcup_{B \subseteq X} (L_B \cdot K^{-1})_A = \bigcup_{B \subseteq X} (L_B \cdot K^{-1}_{A,B})_A$$

Hence, it remains to prove that $\mathrm{Re}((L_B \cdot K^{-1}_{A,B})_A)$ is a recognizable (rational respectively) real trace language. We have

$$\begin{aligned} \mathrm{Re}((L_B \cdot K^{-1}_{A,B})_A) &= \{r \in \mathrm{Re}(\mathbb{C}(X,D)_A) \mid r \cdot \mu_A(y) \in \mathrm{Re}(L_B) \text{ for some } y \in K_{A,B}\} \\ &= \mathrm{Re}(\mathbb{C}(X,D)_A) \cap \mathrm{Re}(L_B) \cdot (\mu_{A,B}(K))^{-1} \end{aligned}$$

Hence, the results follow from Remark 2.2, Lemma 2.4, Propositions 2.13, 2.14 and Lemma 3.4.

The calculus for $K^{-1} \cdot L$ is slightly different. First, note that $K^{-1} \cdot L = \cup_{A \subseteq B \subseteq X} K_A^{-1} \cdot L_B$. Then we have

$$\begin{aligned} K_A^{-1} \cdot L_B &= \{y \in \mathbb{C}(X,D)_{A,B} \mid r\mu_A(y) \in \mathrm{Re}(L_B) \text{ for some } r \in \mathrm{Re}(K_A)\} \\ &= \mathbb{C}(X,D)_{A,B} \cap \{y \in \mathbb{C}(X,D) \mid \mu_A(y) \in \mathrm{Re}(K_A)^{-1} \cdot \mathrm{Re}(L_B)\} \end{aligned}$$

Since by Propositions 2.13 and 2.14, $\mathrm{Re}(K_A)^{-1} \cdot \mathrm{Re}(L_B)$ is recognizable (rational respectively) the results follow from the claim:

Claim: Let $R \subseteq \mathbb{R}(X,D)$ be rational (recognizable respectively) then $M = \{y \in \mathbb{C}(X,D) \mid \mu_A(y) \in R\}$ is rational (recognizable respectively).

For all $B \subseteq X$, we have $\mathrm{Re}(M_B) = \mathrm{Re}(\mathbb{C}(X,D)_B) \cap \{r \in \mathbb{R}(X,D) \mid \mu_A(r) \in R\}$. Therefore, the claim follows directly from

$$\{r \in \mathbb{R}(X,D) \mid \mu_A(r) \in R\} = (\{r \in \mathbb{R}(X,D) \mid \mathrm{alph}(r) \subseteq I(A)\} \cap R) \cdot \mathrm{Suff}_A$$

where Suff_A is the recognizable set defined in Lemma 3.4. $\square$

3.4 Recognizing morphisms

To conclude this section, we will show that recognizable languages of complex traces can be characterized in terms of recognizing morphisms. Of course, since a product of ω finite traces yields always a real trace, we cannot use ω-sequences of finite traces. On the contrary, we have seen that any complex trace can be written as a product of $(\omega+1)$ finite traces. This leads naturally to the following definition: a morphism η from $\mathbb{M}(X,D)$ to a finite monoid S recognizes a subset L of $\mathbb{C}$(X,D) with respect to $(\omega+1)$-sequences if for any $(\omega+1)$-sequence $(u_i)_{i<\omega+1} \subseteq \mathbb{M}(X,D)$, $\prod_{i<\omega+1} u_i \in L$ implies $\prod_{i<\omega+1} \eta^{-1}(\eta(u_i)) \subseteq L$.

Proposition 3.11 *A language $L \subseteq \mathbb{C}(X,D)$ is recognizable if and only if there exists a morphism η from $\mathbb{M}(X,D)$ to a finite monoid S which recognizes L with respect to $(\omega+1)$-sequences.*

Proof: Let η be a morphism from $\mathbb{M}(X,D)$ to a finite monoid S which recognizes L with respect to $(\omega+1)$-sequences. Using Ramsey factorizations (see Section 1.4) it is not difficult to prove that

$$L = \bigcup_{(s,e,s')\in Q} \eta^{-1}(s)\eta^{-1}(e)^{\omega}\eta^{-1}(s')$$

where $Q = \{(s,e,s') \in S^3 / se = s, e^2 = e \text{ and } \eta^{-1}(s)\eta^{-1}(e)^{\omega}\eta^{-1}(s') \subseteq L\}$. For any idempotent e of S, it holds $\eta^{-1}(e) = \eta^{-1}(e)^{+}$, hence $\eta^{-1}(e)^{\omega}$ is recognizable from Theorem 2.7. Therefore, using the closure of Rec $\mathbb{C}$(X,D) under union and concatenation, we deduce that L is recognizable.

In order to prove the converse, let us first start with a remark concerning recognizing morphisms. Let η_1 from $\mathbb{M}(X,D)$ to a finite monoid S_1 and let η_2 from $\mathbb{M}(X,D)$ to a finite monoid S_2 be morphisms recognizing respectively the real trace languages L_1 and L_2. It is easy to verify that the morphism η from $\mathbb{M}(X,D)$ to $S_1 \times S_2$ defined by $\eta(t) = (\eta_1(t), \eta_2(t))$ recognizes both the real trace languages L_1 and L_2.

Now, let L be a recognizable complex trace language. From the remark above, there exists a morphism η from $\mathbb{M}(X,D)$ to a finite monoid S recognizing the real trace languages $\mathrm{Re}(L_A)$ for all $A \subseteq X$.

We construct a finite automaton $(Q, \cdot, q_0, Q)$ such that, for any finite trace t,

$$q_0 \cdot t = (\eta(\mu_B(t)), \mathrm{alph}(\mathrm{Suff}_B(t)))_{B\subseteq X}$$

The finite set of states is $Q = \prod_{B\subseteq X} S \times \mathcal{P}(X)$, the initial state is $q_0 = (1, B)_{B\subseteq X}$ and the transition fonction is given by $(s_B, X_B)_{B\subseteq X} \cdot a = (s'_B, X'_B)_{B\subseteq X}$ where for all $B \subseteq X$, $(s'_B, X'_B) = (s_B \cdot \eta(a), X_B)$ if $a \in I(X_B)$ and $(s'_B, X'_B) = (s_B, X_B \cup \{a\})$ otherwise. Let T be the transition monoid of this automaton, the transition morphism ψ from $\mathbb{M}(X, D)$ in T satisfies for all $s, t \in \mathbb{M}(X, D)$ and $B \subseteq X$ the following properties:

$$\psi(s) = \psi(t) \Longrightarrow \eta(\mu_B(s)) = \eta(\mu_B(t))$$

$$\psi(s) = \psi(t) \Longrightarrow \mathrm{alph}(\mathrm{Suff}_B(s)) = \mathrm{alph}(\mathrm{Suff}_B(t))$$

Note that the morphism ψ recognizes the languages $\mathrm{Re}(L_A)$ for all $A \subseteq X$, too. Indeed, the first implication above, in the particular case where $B = \emptyset$, proves that $\eta(s) = \eta(t)$ as soon as $\psi(s) = \psi(t)$. Moreover, using the second implication with $B = X$, we deduce that the morphism ψ preserves the alphabet, i.e. $\mathrm{alph}(s) = \mathrm{alph}(t)$ as soon as $\psi(s) = \psi(t)$.

We claim that the morphism ψ recognizes the complex trace language L with respect to $(\omega + 1)$-sequences.

Let $(u_i)_{i<\omega+1}, (v_i)_{i<\omega+1} \subseteq \mathbb{M}(X, D)$ be two $(\omega+1)$-sequences such that, for all $i < \omega + 1$, $\psi(u_i) = \psi(v_i)$ and $\prod_{i<\omega+1} u_i \in L$. Let $B = \mathrm{alphinf}(\prod_{i<\omega} u_i)$ and $D(A) = \mathrm{Im}(\prod_{i<\omega+1} u_i) = D(B) \cup D(\mathrm{Suff}_B(u_\omega))$. From the properties of ψ, we deduce that $\mathrm{alphinf}(\prod_{i<\omega} v_i) = B$ and $\mathrm{Im}(\prod_{i<\omega+1} v_i) = D(B) \cup D(\mathrm{Suff}_B(v_\omega)) = D(A)$. Hence, $\mathrm{Re}(\prod_{i<\omega+1} u_i) \in \mathrm{Re}(L_A)$ and it remains to prove that $\mathrm{Re}(\prod_{i<\omega+1} v_i) \in \mathrm{Re}(L_A)$, too. Now, there exists some integer n such that

$$\mathrm{Re}(\prod_{i<\omega+1} u_i) = (\prod_{i<\omega} u_i) \cdot \mu_B(u_\omega) = (\prod_{i<n} u_i) \cdot \mu_B(u_\omega) \cdot (\prod_{n\le i<\omega} u_i)$$

and

$$\mathrm{Re}(\prod_{i<\omega+1} v_i) = (\prod_{i<\omega} v_i) \cdot \mu_B(v_\omega) = (\prod_{i<n} v_i) \cdot \mu_B(v_\omega) \cdot (\prod_{n\le i<\omega} v_i)$$

From the definition of ψ, we have $\eta(u_i) = \eta(v_i)$ for all $i < \omega$ and $\eta(\mu_B(u_\omega)) = \eta(\mu_B(v_\omega))$. This concludes the proof since η recognizes $\mathrm{Re}(L_A)$ with respect to ω-sequences. $\square$

A natural question which arises from Proposition 3.11 is whether this result holds for arbitrary ordinals. The answer is yes as we will sketch below. We do not give complete proofs, since for complex traces higher ordinals than $\omega + 1$ are not necessary and of less importance. Readers not familiar with ordinals may skip the end of this section without loss for the rest of the paper.

First, we may convince ourselves that for any ordinal β (countable or not) and for any β-sequence $(u_i)_{i<\beta}$ of complex traces its product $u = \prod_{i<\beta} u_i$ is a well-defined complex trace. This can be seen e.g. by transfinite induction or by an adequate use of dependence graph representations (which allows, if necessary, uncountable vertex sets).

We then say that a language $L \subseteq \mathbb{C}(X, D)$ is recognized by a morphism η from $\mathbb{M}(X, D)$ to a finite monoid S *with respect to all ordinal sequences*, whenever $(u_i)_{i<\beta}, (v_i)_{i<\beta}$ are ordinal sequences such that $\prod_{i<\beta} u_i \in L$ and $\eta(u_i) = \eta(v_i)$ for all $i < \beta$ then we have $\prod_{i<\beta} v_i \in L$.

We have the following proposition:

Proposition 3.12 *A complex trace language $L \subseteq \mathbb{C}(X, D)$ is recognizable if and only if some morphism η from $\mathbb{M}(X, D)$ to a finite monoid S recognizes L with respect to all ordinal sequences.*

Since $(\omega+1)$-sequences are special cases of all ordinal sequences, one direction of the assertion above is clear from Proposition 3.11. For the other direction, one reduces first to the case of ordinals $\beta \leq \omega \cdot |X|$. Then one uses basically the same idea as in the case of $(\omega+1)$-sequences of merging a β-sequence with $\beta \leq \omega \cdot |X|$ into one ω sequence for the real part. Furthermore, η can be choosen such that from its alphabetical information the imaginary part of a product $\prod_{i<\beta} u_i$ is known from the sequence $(\eta(u_i))_{i<\beta}$ of elements of the monoid S.

We can also prove that for any ordinal β there exists some ordinal α such that $\alpha < \omega(|X|+1)$ and $L^\beta = L^\alpha$ for all subsets $L \subseteq \mathbb{C}(X,D)$. Again, details (which are in fact not obvious) are left to the reader. Due to this fact, we can derive the following further result:

Proposition 3.13 *Let β be any ordinal and $L \subseteq \mathbb{C}(X,D)$.*
If L and L^+ are both recognizable, then L^β and $\bigcup_{\alpha<\beta} L^\alpha$ are recognizable, too.
If L is rational, then L^β and $\bigcup_{\alpha<\beta} L^\alpha$ are rational, too.

Recognizable sets of transfinite words have been defined using automata on transfinite words, see for instance (Büchi, 1965; Büchi, 1973; Wojciechowski, 1984; Wojciechowski, 1985). We should warn the reader that a language of transfinite words recognized by a morphism with respect to all ordinal sequences is not necessarily recognizable by transfinite automata. For instance, the alphabetic morphism alph : $X^* \longrightarrow \mathcal{P}(X)$ recognizes the language $\{a^\alpha\}$ with respect to all ordinal sequences for all limit ordinals α and all letters $a \in X$. However, such languages are recognizable by transfinite automata if and only if $\alpha < \omega^\omega$ (Wojciechowski, 1984). Hence, this notion of recognizability by morphisms with respect to all ordinal sequences may only be suitable in the special case of complex traces. Indeed, for each β-sequence $(u_i)_{i\in\beta}$ of complex traces, there exists a subsequence $(u_i)_{i\in\alpha}$ (with $\alpha \subseteq \beta$) of length at most $\omega \cdot |X|$ such that $\prod_{i\in\beta} u_i = \prod_{i\in\alpha} u_i$. Therefore, the problem caused by sequences of length higher than ω^ω is not relevant for complex traces.

4 C-rational complex trace languages

We end, in this section, the study of the family of recognizable complex trace languages. We extend the theorems of Büchi and Ochmański to complex traces by proving that the recognizable complex trace languages are exactly the c-rational ones (Theorem 4.7). To this purpose, we define a suitable notion of connected complex trace and we prove, as an intermediary result, that the iteration of a recognizable language of connected complex traces remains recognizable (Theorem 4.3).

4.1 Connected complex traces

In order to define concurrent iterations for complex trace languages, we need first a notion of connectedness for complex traces. Intuitively, a trace is connected if it cannot be split into two non-empty independent traces. Thus, let us introduce the notion of independence. This notion is clear in $\mathbb{G}(X,D)$: two graphs f and g are independent if $\mathrm{alph}(f)\times\mathrm{alph}(g) \subseteq I$. Unfortunately, we cannot use this definition in $\mathbb{C}(X,D)$ since the alphabet of a complex trace is not well-defined. Nevertheless, it is easy to verify that the independence relation in $\mathbb{G}(X,D)$ factorizes to complex traces. Hence the independence relation in $\mathbb{C}(X,D)$ is well-defined and can be characterized as follows. Two complex traces $x = (u, D(A))$, $y = (v, D(B)) \in \mathbb{C}(X,D)$ are *independent* if $(\mathrm{alph}(u) \cup A) \times (\mathrm{alph}(v) \cup B) \subseteq I$, in this case we will simply write $(x,y) \in I$. Note that this is well-defined since for all $E, F \subseteq X$ it holds $E \times F \subseteq I \iff E \cap D(F) = \emptyset$. This leads to the following definition.

Definition: *A complex trace z is called* connected *if for all factorization $z = xy$ where x, y are independent, either $x = 1$ or $y = 1$. A trace x is called* a connected component *of $z \in \mathbb{C}(X,D)$ if either $x = z = 1$ or x is a non-empty connected trace such that $z = xy$ for some y independent of x. The trace z is said to be* finitary connected *if it admits at most one finite connected component.*

For any non-empty trace z, we will denote by $C(z)$ (respectively $\mathrm{FC}(z) = C(z) \cap \mathbb{M}(X,D)$) the set of connected components (respectively finite connected components) of z. These notations are as usual extended to languages. A language is said to be connected (respectively finitary connected) if all its elements are connected (respectively finitary connected).

For instance, let $(X,D) = a — b — c — d — f — g$ (with e adjacent to b and c), the complex traces $(gga^\omega, D(\{a\}))$ and $(d^\omega, D(\{c,d\}))$ are independent whereas the traces $(gga^\omega, D(\{a,b\}))$ and $(d^\omega, D(\{c,d\}))$ are not. The trace $x = (gg(ad)^\omega, D(\{a,c,d\}))$ is not a connected trace since $x = (gga^\omega, D(\{a\})) \cdot (d^\omega, D(\{c,d\}))$ is the product of two independent traces. The connected component of x are $(gg, \emptyset), (a^\omega, D(\{a\})), (d^\omega, D(\{c,d\})), (a^\omega, D(\{a,b\}))$ and $(d^\omega, D(\{d\}))$.

Note that a direct translation of the above definition can be used to define connectedness notions on dependence graphs as well. These notions coincide with the natural notion of connectedness in $\mathbb{G}(X,D)$: a dependence graph is connected if and only if so is its underlying undirected graph. Unfortunately, this natural notion does not factorize to complex traces. The same complex trace might have connected as well as disconnected representing dependence graphs. For example let $(X,D) = a — b — c$. Then the complex trace $x = ((ac)^\omega, X)$ is represented by the connected dependence graph $(ac)^\omega b$ as well as by the disconnected graph $a^\omega c^\omega$. This explains why we define the notion of connected traces through the independence relation. For finitary connectedness the situation is simpler. A trace $x \in \mathbb{M}(X,D)$ is a finite connected component of $z \in \mathbb{C}(X,D)$ if and only if x is finite and a connected component of any representing dependence graph of z. The following proposition describes the links between the two notions of connectedness in $\mathbb{G}(X,D)$ and $\mathbb{C}(X,D)$.

Proposition 4.1 *Let $x \in \mathbb{C}(X,D)$ be a complex trace.*

i) x is connected if and only if all its representatives in $\mathbb{G}(X,D)$ are connected.

ii) Let $f \in \mathbb{G}(X,D)$ be any representative of x and let $\{f_1, \ldots, f_k\}$ be the set of finite connected components of f. Then $\mathrm{FC}(x) = \{f_1, \ldots, f_k\}$.

Proof: i) Let f be a representative of x in $\mathbb{G}(X,D)$. Suppose that $f = gh$ with $\mathrm{alph}(g) \times \mathrm{alph}(h) \subseteq I$. Let y, z be the complex traces represented by g, h respectively. Hence $x = yz$ with $(y,z) \in I$. Since x is connected, we get $y = 1$ or $z = 1$ and thus $g = 1$ or $h = 1$. Therefore f is connected.
Conversely, suppose that $x = yz$ with $(y,z) \in I$. Let g, h be representatives in $\mathbb{G}(X,D)$ of the complex traces y, z respectively. Hence gh is a representative of x and $(g,h) \in I$. Thus $g = 1$ or $h = 1$, hence $y = 1$ or $z = 1$ and x is connected.
ii) Let g be a finite connected component of f, $f = gh$ for some $h \in \mathbb{G}(X,D)$ with $\mathrm{alph}(g) \times \mathrm{alph}(h) \subseteq I$. Let y be the complex trace represented by h. Since g is finite, we get $x = gy$ with $(g,y) \in I$. Moreover g is a connected complex trace from i). Hence $g \in \mathrm{FC}(x)$.
Conversely, let $y \in \mathrm{FC}(x)$. Then $x = yz$ for some complex trace z such that $(y,z) \in I$. Since $\mathbb{G}(X,D)$ is left cancellative (Diekert, 1991), the dependence graph $y^{-1}f$ is well-defined and is a representative of the complex trace z. Hence y is a finite connected component of $f = y(y^{-1}f)$. $\square$

4.2 Concurrent iterations

In order to generalize Büchi's and Ochmanski's Theorems to complex traces, we need a link to the language $\nu_A(L) = \mu_{A,A}(L)$ introduced at the begining of Section 3. For a complex trace x, any non-empty product over pairwise independent finite connected components of x is called a *finite component* of x. For a language $L \subseteq \mathbb{C}(X,D)$, the set of finite components of the elements of L is denoted by $\mathrm{FinComp}(L)$.

Lemma 4.2 *Let $L \subseteq \mathbb{C}(X,D)$ be recognizable. Then the following assertions hold.*

i) $\mathrm{FinComp}(L) = \bigcup_{A \subseteq X} \nu_A(L)$

ii) $\mathrm{FC}(L)$ *and* $\mathrm{C}(L)$ *are recognizable*

Proof: i) Let $t \in \nu_A(L)$. By definition, we have $\mathrm{alph}(t) \subseteq I(A)$ and $\mathrm{Im}(t) \subseteq D(A)$, hence $\mathrm{Im}(t) = \emptyset$ and thus t is finite. Moreover, there exists some $x \in L_{A,A}$ such that $t = \mu_A(x)$. Hence $x = t \cdot (r, \mathrm{Im}(x))$ and, since $x \in L_{A,A}$, we have $(D(\mathrm{alph}(r)) \cup \mathrm{Im}(x)) \subseteq D(A)$. Thus, from $\mathrm{alph}(t) \subseteq I(A)$, $(t, (r, \mathrm{Im}(x))) \in I$. Therefore $t \in \mathrm{FinComp}(L_{A,A}) \subseteq \mathrm{FinComp}(L)$.

Conversely, let $t \in \mathrm{FinComp}(L)$, there exist some $u \in L$ and $v \in \mathbb{C}(X,D)$ such that $u = tv$ with $(t,v) \in I$. Let $B = \mathrm{alph}(t)$ and $A = I(B)$, note that $B \subseteq I(A)$. We claim that $t = \mu_A(u)$ and $u \in L_{A,A}$ whence $t \in \nu_A(L)$. Let $v = (r, D(C))$, since $(t,v) \in I$, we have by definition $B \times (\mathrm{alph}(r) \cup C) \subseteq I$, and thus $(\mathrm{alph}(r) \cup C) \subseteq I(B) = A$. Hence $\mathrm{alph}(r) \subseteq A$, in particular $\mathrm{alph}(r) \cap I(A) = \emptyset$ and then $\mu_A(u) = t$. We get also $C \subseteq A$, hence $u \in L_{A,A}$ and the claim is proved.

ii) Let NC be the set of non-connected complex traces. It is easy to verify that $NC = \bigcup_{(A \cup B) \times (E \cup F) \subseteq I} K(A,B) \cdot K(E,F)$ where, for all $A, B \subseteq X$, $K(A,B) = \{x \in \mathbb{C}(X,D)_B \setminus \{1\} \mid \mathrm{alph}(\mathrm{Re}(x)) = A\}$. Now, it turns out that

$$\mathrm{C}(L) = (\bigcup_{(A \cup B) \times (E \cup F) \subseteq I} (K(A,B)^{-1} \cdot L) \cap K(E,F)) \cap (\mathbb{C}(X,D) \setminus NC)$$

Clearly, for all $A, B \subseteq X$, the set $K(A,B)$ is recognizable. From Proposition 3.1 and Theorems 3.5 and 3.10, we deduce that NC and $\mathrm{C}(L)$ are recognizable. Thus the set $\mathrm{FC}(L) = \mathrm{C}(L) \cap \mathbb{M}(X,D)$ is recognizable, too.□

Theorem 4.3 *Let $L \subseteq \mathbb{C}(X,D)$ be recognizable. If L is finitary-connected (in particular if L is connected) or if $\mathrm{FC}(L) \subseteq L^*$, then L^* and L^ω are recognizable.*

Proof: If L is finitary connected, then $\mathrm{FinComp}(L)$ is connected and from Lemma 4.2 i), it holds that $\nu_A(L)$ is connected for all $A \subseteq X$. Since by Lemma 3.4 $\nu_A(L)$ is also recognizable, $\nu_A(L)^*$ is recognizable by (Métivier, 1986; Ochmanski, 1985). Hence L^* and L^ω are recognizable by Corollary 3.9.

Assume that $\mathrm{FC}(L) \subseteq L^*$. We will also show that $\nu_A(L)^*$ is recognizable for all $A \subseteq X$. For this purpose, we claim that $\nu_A(L)^* = \nu_A(\mathrm{FC}(L))^*$.

Remark first that if t is a finite connected trace, from Lemma 4.2 i) applied to $L = \{t\}$, we get that, for any $A \subseteq X$, either $\nu_A(t) = \{t\}$ if $\mathrm{alph}(t) \subseteq I(A)$ or $\nu_A(t) = \emptyset$ otherwise.

Let now t be in $\nu_A(L)$ and $\{t_1, \ldots, t_k\}$ be the connected components of t. From Lemma 4.2 i), we have $\{t_1, \ldots, t_k\} \subseteq \mathrm{C}(\mathrm{FinComp}(L)) = \mathrm{FC}(L)$. From the remark above, for all $1 \leq i \leq k$, we get $\{t_i\} = \nu_A(t_i)$ and thus $t_i \in \nu_A(\mathrm{FC}(L))$. Hence $\nu_A(L) \subseteq \nu_A(\mathrm{FC}(L))^*$, and $\nu_A(L)^* \subseteq \nu_A(\mathrm{FC}(L))^*$, too.

Conversely, since $\mathrm{FC}(L) \subseteq L^*$, we have $\nu_A(\mathrm{FC}(L)) \subseteq \mathrm{FinComp}(\mathrm{FC}(L)) = \mathrm{FC}(L) \subseteq L^*$. Let $t \in \nu_A(\mathrm{FC}(L)) \subseteq L^*$, in particular $\mathrm{alph}(t) \subseteq I(A)$ and $t = t_1 \ldots t_k$ with $t_i \in L_\emptyset$ and $\mathrm{alph}(t_i) \subseteq I(A)$ for $1 \leq i \leq k$. From the remark above, $\{t_i\} = \nu_A(t_i) \subseteq \nu_A(L)$, for $1 \leq i \leq k$. Therefore $t \in \nu_A(L)^*$ and $\nu_A(\mathrm{FC}(L)) \subseteq \nu_A(L)^*$. Finally $\nu_A(L)^* = \nu_A(\mathrm{FC}(L))^*$ and the claim is proved.

Since $\nu_A(\mathrm{FC}(L))$ is connected, we obtain from (Métivier, 1986; Ochmanski, 1985) that $\nu_A(L)^*$ is recognizable. □

Corollary 4.4 *Let $L \subseteq \mathbb{C}(X,D)$ be recognizable, then $(\mathrm{FC}(L) \cup L)^*$ and $(\mathrm{FC}(L) \cup L)^\omega$ are recognizable, too.*

Proof: This is immediate from the theorem above and the observation that we have $\mathrm{FC}(\mathrm{FC}(L) \cup L) = \mathrm{FC}(L)$. □

Beside its own interest, Corollary 4.4 yields a simple proof of the following result which is surprising, since the $*$-iteration of a single trace is not recognizable, in general.

Corollary 4.5 *Let $x \in \mathbb{C}(X,D)$ be a recognizable complex trace, i.e., $\{x\} \in \mathrm{Rec}(\mathbb{C}(X,D))$, then x^ω is recognizable.*

Proof: Choose $A \subseteq X$ such that $D(A) = \mathrm{Im}(x^\omega)$. Then we claim that

$$\{x^\omega\} = ((\mathrm{FC}(\{x\}) \cup \{x\})^\omega)_A$$

One inclusion is obvious. Conversely, let $z \in ((\mathrm{FC}(\{x\}) \cup \{x\})^\omega)_A$. Since $\mathrm{Im}(z) = \mathrm{Im}(x^\omega) = D(\mathrm{alph}(\mathrm{Re}(x))) \cup \mathrm{Im}(x)$, we deduce that $z = zx = zx^\omega$. Now, let $\mathrm{FC}(\{x\}) =$

$\{x_1, \ldots, x_k\}$ and let $y = (x_1 \ldots x_k)^{-1} \cdot x$. From the definition of connected components, it holds that $x_1, \ldots, x_k, y$ are pairwise independent. Therefore $z = x_1^{n+n_1} \ldots x_k^{n+n_k} y^n$ for some $0 \leq n_1, \ldots, n_k, n \leq \omega$, $x^\omega = x_1^\omega \ldots x_k^\omega y^\omega$ and hence $zx^\omega = x^\omega$. $\square$

Now, we extend the concurrent iteration of Ochmanski (Ochmanski, 1985) to complex traces.

Definition: *Let $L \subseteq \mathbb{C}(X, D)$. Then the concurrent iterations of L are defined by $L^{c-*} = (\mathrm{C}(L))^*$ and $L^{c-\omega} = (\mathrm{C}(L))^\omega$.*

Since $\mathrm{C}(L)$ is connected, we obtain from Lemma 4.2 and Theorem 4.3:

Corollary 4.6 *Let $L \subseteq \mathbb{C}(X, D)$ be recognizable. Then L^{c-*} and $L^{c-\omega}$ are recognizable.*

Definition: *The family of c-rational complex trace languages, $\mathrm{c}-\mathrm{Rat}(\mathbb{C}(X, D))$, is the smallest family which contains all finite sets of finite traces and which is closed under the operations union, concatenation and concurrent iterations c-$*$ and c-ω.*

We are now ready to state our main result.

Theorem 4.7 $\mathrm{Rec}(\mathbb{C}(X, D)) = \mathrm{c} - \mathrm{Rat}(\mathbb{C}(X, D))$

Proof: We have seen that $\mathrm{Rec}(\mathbb{C}(X, D))$ is a boolean algebra closed under concatenation (Theorem 3.5) and concurrent iterations (Corollary 4.6). Hence $\mathrm{c} - \mathrm{Rat}(\mathbb{C}(X, D)) \subseteq \mathrm{Rec}(\mathbb{C}(X, D))$. Conversely, we have seen (Corollary 3.6) that every recognizable language can be written as a finite union over languages of type $MN_1^\omega \ldots N_k^\omega s$ where $M, N_1, \ldots, N_k \subseteq \mathbb{M}(X, D)$ are recognizable and $s \in \mathbb{M}(X, D)$ is a finite trace. Moreover all traces in N_i have the same connected alphabet and hence $N_i^\omega = N_i^{c-\omega}$. From Ochmanski's theorem (Ochmanski, 1985), the recognizable languages of finite traces M and N_i, for $1 \leq i \leq k$, are in $\mathrm{c} - \mathrm{Rat}(\mathbb{C}(X, D))$. Therefore L is also in $\mathrm{c} - \mathrm{Rat}(\mathbb{C}(X, D))$. $\square$

Remark 4.8 i) Every recognizable complex trace language has effectively a c-rational expression where the concurrent iterations are allowed only for languages of finite traces which are connected. (Hence where the concurrent iterations become the usual Kleene-$*$ and ω-iteration.)

ii) We could have defined the finitary concurrent iteration of a language L as $L^{fc-\dagger} = (\mathrm{FC}(L) \cup L)^\dagger$ for $\dagger \in \{*, \omega\}$. An analogue of Theorem 4.7 with a suitable definition of $\mathrm{fc} - \mathrm{Rat}(\mathbb{C}(X, D))$ also holds in this case.

5 Conclusion

In this paper we have studied the families of rational and recognizable languages of complex traces. We showed that classical results on finite and infinite words and finite traces have a natural extension to complex traces. In our investigations we followed mainly an algebraic viewpoint with the final goal to establish the Kleene-Ochmanki-Theorem.

Of course, other characterizations of the recognizable languages are of high interest. For instance, word languages can be defined by monadic second order logic (Büchi, 1962). The equivalent work for finite traces can be found in (Thomas, 1990b) and for real traces in (Ebinger and Muscholl, 1993). Its generalization to complex traces seems to be possible. Besides, the most well-known characterizations of recognizable languages use the notion of finite automata. Recognizable word languages are languages which are accepted by (non-deterministic) Büchi automata or by deterministic Muller automata. Asynchronous (cellular) automata (Zielonka, 1987; Zielonka, 1989) provide distributed acceptors for recognizable languages of finite traces. With a suitable Büchi like condition, non-deterministic asynchronous (cellular) automata accept exactly the recognizable languages of real traces (Gastin and Petit, 1992) and may be easily extended to accept complex trace languages. Finally, a characterization of recognizable real trace languages with deterministic Muller asynchronous (cellular) automata was recently proved in (Diekert and Muscholl, 1993).

Acknowledgements: We thank the anonymous referees for the useful suggestions and comments which in particular led to Section 3.4. We also thank Anca Muscholl for fruitful discussions.

References

Aalbersberg, I. and Rozenberg, G. (1988). Theory of traces. *Theoret. Comput. Sci.*, 60:1–82.

Arnold, A. (1985). A syntactic congruence for rational ω-languages. *Theoret. Comput. Sci.*, 39:333–335.

Bonizzoni, P., Mauri, G., and Pighizzini, G. (1990). About infinite traces. In Diekert, V., editor, *Proceedings of the ASMICS workshop Free Partially Commutative Monoids, Kochel am See (Germany) 1989*, Report TUM-I9002, Technical University of Munich, pages 1–10.

Büchi, J. (1962). On a decision method in restricted second order arithmetic. In *Proc. Internat. Congress on Logic, Methodology and Philosophy*, pages 1–11. Standford University Press.

Büchi, J. (1965). Decision methods in the theory of ordinals. *Bull. of the AMS*, 71:767–770.

Büchi, J. (1973). *The monadic second order theory of* ω_1. Lecture Notes in Mathematics 328. Springer, Berlin-Heidelberg-New York.

Cartier, P. and Foata, D. (1969). *Problèmes combinatoires de commutation et réarrangements.* Lecture Notes in Mathematics 85. Springer, Berlin-Heidelberg-New York.

Clerbout, M. and Latteux, M. (1987). Semi-Commutations. *Inform. and Comput.*, 73:59–74.

Cori, R. and Perrin, D. (1985). Automates et commutations partielles. *R.A.I.R.O.-Informatique Théorique et Applications*, 19:21–32.

Diekert, V. (1990). *Combinatorics on Traces*. Lecture Notes in Computer Science 454. Springer, Berlin-Heidelberg-New York.

Diekert, V. (1991). On the concatenation of infinite traces. In Choffrut, C. et al., editors, *Proceedings of the 8th Annual Symposium on Theoretical Aspects of Computer Science (STACS'91), Hamburg (Germany) 1991*, Lecture Notes in Computer Science 480, pages 105–117. Springer, Berlin-Heidelberg-New York. To appear in TCS.

Diekert, V., Gastin, P., and Petit, A. (1991). Recognizable complex trace languages. In Tarlecki, A., editor, *Proceedings of the 16th Symposium on Mathematical Foundations of Computer Science (MFCS'91), Kazimierz Dolny (Poland) 1991*, Lecture Notes in Computer Science 520, pages 131–140. Springer, Berlin-Heidelberg-New York.

Diekert, V. and Muscholl, A. (1993). Deterministic asynchronous automata for infinite traces. In Enjalbert, P., Finkel, A., and Wagner, K. W., editors, *Proceedings of the 10th Annual Symposium on Theoretical Aspects of Computer Science (STACS'93), Würzburg 1993*, number 665 in Lecture Notes in Computer Science, pages 617–628. Springer, Berlin-Heidelberg-New York.

Ebinger, W. and Muscholl, A. (1993). On logical definability of ω-trace languages. In *Proceedings of the 20th International Colloquium on Automata Languages and Programming (ICALP'93), Lund (Sweden) 1993*, Lecture Notes in Computer Science. Springer, Berlin-Heidelberg-New York. To appear.

Eilenberg, S. (1974). *Automata, Languages and Machines*, volume I. Academic Press, New York and London.

Flé, M. and Roucairol, G. (1982). On the serializability of iterated transactions. In *Proceedings ACM SIGACT-SIGOPS, Symposium on Principles of Distr. Comp., Ottawa (Canada) 1982*, pages 194–200.

Fliess, M. (1974). Matrices de Hankel. *J. Math. Pures et Appl.*, 53:197–224.

Gastin, P. (1990). Infinite traces. In Guessarian, I., editor, *Proceedings of the Spring School of Theoretical Computer Science on Semantics of Systems of Concurrent Processes*, Lecture Notes in Computer Science 469, pages 277–308. Springer, Berlin-Heidelberg-New York.

Gastin, P. (1991). Recognizable and rational trace languages of finite and infinite traces. In Choffrut, C. et al., editors, *Proceedings of the 8th Annual Symposium on Theoretical Aspects of Computer Science (STACS'91), Hamburg (Germany))1991*, Lecture Notes in Computer Science 480, pages 89–104. Springer, Berlin-Heidelberg-New York.

Gastin, P., Ochmański, E., Petit, A., and Rozoy, B. (1992). Decidability of the star problem in $A^* \times \{b\}^*$. *Information Processing Letters*, 44:65–71.

Gastin, P. and Petit, A. (1992). Asynchronous cellular automata for infinite traces. In Kuich, W., editor, *Proceedings of the 19th International Colloquium on Automata Languages and Programming (ICALP'92), Vienna (Austria) 1992*, volume 623 of *Lecture Notes in Computer Science*, pages 583–594. Springer, Berlin-Heidelberg-New York.

Gastin, P., Petit, A., and Zielonka, W. (1991). A Kleene theorem for infinite trace languages. In Leach Albert, J. et al., editors, *Proceedings of the 18th International Colloquium on Automata Languages and Programming (ICALP'91), Madrid (Spain) 1991*, Lecture Notes in Computer Science 510, pages 254–266. Springer, Berlin-Heidelberg-New York.

Gastin, P. and Rozoy, B. (1991). The poset of infinitary traces. Technical Report, LITP 91.24, Université de Paris 6. To appear in *Theoret. Comp. Sci.*

Graham, R. (1981). Rudiments of Ramsey theory. Regional Conference Series in Mathematics 45.

Kwiatkowska, M. (1990). A metric for traces. *Imform. Proc. Letters*, 35:129–135.

Mazurkiewicz, A. (1977). Concurrent program schemes and their interpretations. DAIMI Rep. PB 78, Aarhus University, Aarhus.

Mazurkiewicz, A. (1987). Trace theory. In Brauer, W. et al., editors, *Petri Nets, Applications and Relationship to other Models of Concurrency*, Lecture Notes in Computer Science 255, pages 279–324. Springer, Berlin-Heidelberg-New York.

Métivier, Y. (1986). Une condition suffisante de reconnaissabilité dans un monoïde partiellement commutatif. *R.A.I.R.O.-Informatique Théorique et Applications*, 20:121–127.

Métivier, Y. and Rozoy, B. (1991). On the star operation in free partially commutative monoids. *International Journal of Foundations of Computer Science*, 2:257–265.

Ochmanski, E. (1985). Regular behaviour of concurrent systems. *Bull. of the European Association for Theoretical Computer Science (EATCS)*, 27:56–67.

Ochmanski, E. (1990). Notes on a star mystery. *Bull. of the European Association for Theoretical Computer Science (EATCS)*, 40:252–257.

Perrin, D. (1989). Partial commutations. In Ausiello, G. et al., editors, *Proceedings of the 16th International Colloquium on Automata, Languages and Programming (ICALP'89), Stresa (Italy) 1989*, Lecture Notes in Computer Science 372, pages 637–651. Springer, Berlin-Heidelberg-New York.

Perrin, D. and Pin, J. (1991). Mots Infinis. Technical Report, LITP 91.06, Université de Paris 6.

Sakarovitch, J. (1987). On regular trace languages. *Theoret. Comput. Sci.*, 52:59–75.

Thomas, W. (1990a). Automata on infinite objects. In Leeuwen, J., editor, *Handbook on Theoretical Computer Science*, Vol. B, pages p. 135–191.

Thomas, W. (1990b). On logical definability of trace languages. In Diekert, V., editor, *Proceedings of the ASMICS workshop Free Partially Commutative Monoids, Kochel am See, Oktober 1989*, Report TUM-I9002, Technical University of Munich, pages 1–10.

Wojciechowski, J. (1984). Classes of transfinite sequences accepted by non-deterministic finite automata. *Fundamenta Informaticae*, 7:191–223.

Wojciechowski, J. (1985). Finite automata on transfinite sequences and regular expressions. *Fundamenta Informaticae*, 8:379–396.

Zielonka, W. (1987). Notes on finite asynchronous automata. *R.A.I.R.O.-Informatique Théorique et Applications*, 21:99–135.

Zielonka, W. (1989). Safe executions of recognizable trace languages by asynchronous automata. In Mayer, A. R. et al., editors, *Proceedings Symposium on Logical Foundations of Computer Science, Logic at Botik '89, Pereslavl-Zalessky (USSR) 1989*, Lecture Notes in Computer Science 363, pages 278–289. Springer, Berlin-Heidelberg-New York.