A KLEENE THEOREM FOR INFINITE TRACE LANGUAGES*

Paul GASTIN
Université PARIS 6
LITP, Institut Blaise Pascal
4, place Jussieu
75 252 PARIS CEDEX 05
FRANCE

Antoine PETIT
Université PARIS SUD
LRI, URA CNRS 410
Bât. 490
91 405 ORSAY CEDEX
FRANCE

Wieslaw ZIELONKA
Université BORDEAUX 1
LaBRI
351, Cours de la Libération
33 405 TALENCE CEDEX
FRANCE

Abstract : Kleene's theorem is considered as one of the cornerstones of theoretical computer science. It ensures that, for languages of finite words, the family of recognizable languages is equal to the family of rational languages. It has been generalized in various ways, for instance, to formal power series by Schützenberger, to infinite words by Büchi and to finite traces by Ochmanski. Finite traces have been introduced by Mazurkiewicz in order to modelize the behaviours of distributed systems. The family of recognizable trace languages is not closed by Kleene's star but by a concurrent version of this iteration. This leads to the natural definition of co-rational languages obtained as the rational one by simply replacing the Kleene's iteration by the concurrent iteration. Cori, Perrin and Métivier proved, in substance, that any co-rational trace language is recognizable. Independently, Ochmanski generalized Kleene's theorem showing that the recognizable trace languages are exactly the co-rational languages. Besides, infinite traces have been recently introduced as a natural extension of both finite traces and infinite words. In this paper we generalize Kleene's theorem to languages of infinite traces proving that the recognizable languages of finite or infinite traces are exactly the co-rational languages.

1. Introduction

The characteristic property of asynchronous distributed systems is the absence of any kind of centralized control mechanism. The actions executed by separate components are causally independent and different external observers can witness different time ordering of their execution, in the same computation. Thus, when we specify or examine the behaviour of a parallel system, the order in which independent actions are executed seems irrelevant and even impossible to precise. For these reasons, Mazurkiewicz [Maz77] proposed to identify two sequential behaviours if they differ only in the order of independent actions. This induces an equivalence relation over the set of sequences of actions and Mazurkiewicz coined the name traces for its equivalence classes. For a fixed independence relation, traces form a monoid. These monoids were first considered by combinatorists [CF69], but since they provide a natural framework to describe concurrent behaviours, they have been intensively studied by computer scientists in the last years. This approach have led to new and very fruitful research directions. See, for instance, the monograph [Die90] or surveys [Maz86], [AR88], [Per89], where also extensive references of the subject are given.
Since recognizable trace languages describe the behaviour of finite state systems, they constitute the basic family of trace languages. One of the most important problem in trace theory was to find an analogue of Kleene's theorem. Cori and Perrin [CP85] proved that the family of

* This work has been partly supported by the ESPRIT Basic Research Actions N° 3166 (ASMICS) and N° 3148 (DEMON) and by the PRC Math-Info.

recognizable trace languages is closed under boolean operations and concatenation. Unfortunately, it turns out that there exist recognizable trace languages T such that T* is not recognizable. Several papers discussed the recognizability of T* [FR85], [Sak87], [Och90], [Roz90], [MR91]. But the most interesting and useful sufficient condition has been found by Métivier [Met86]. He showed that if a recognizable language T consists of connected traces then the language T* remains recognizable. Independently, Ochmanski [Och85] defined a new natural operation on trace languages, the concurrent iteration, which iterates independently the connected components of a trace language. This leads to the natural definition of co-rational languages obtained as the rational one by simply replacing the Kleene's iteration by the concurrent iteration. Ochmanski, generalizing Kleene's theorem, settled positively the problem stated above. He proved that the recognizable trace languages coincide with the co-rational trace languages.
Besides this theory of finite traces, a theory of infinite traces has recently begun in order to modelize infinite behaviours of distributed systems. A definition of an equivalence between infinite words, which generalizes the one on finite words refered above, can be found in [FR85] and [Gas88]. Intuitive ideas on infinite dependence graphs, which are another representation of traces, are proposed in [Maz86]. Then, this theory has been develloped in several papers. Topological properties of infinite traces [Gas90], [BMP89], PoSet properties [Kwi89], [GR91] and links with event structures [GR91] have in particular been studied. Using a topological completion of the finite trace monoid, Diekert [Die91] proposed a nice generalization of infinite traces to "complex infinite traces".

We are interested here in the family of recognizable languages of infinite traces introduced in [Gas90]. This family is closed under boolean operations and concatenation but, similarly to the case of finite traces, it is closed neither under Kleene's iteration nor infinite iteration [Gas91].
In this paper, we prove that, with suitable definitions, there is a Kleene like characterization which applies to infinite traces. Therefore, we generalize both Ochmanski's result on finite traces and Büchi's result on infinite words. In other words, we suppress the question mark in the diagram below.

	Words	Traces
Finite	Kleene	Ochmanski
Infinite	Büchi	?

Extending the ideas of Ochmanski, we define concurrent infinite iteration. We show that co-rational expressions, obtained from the usual rational expressions by replacing the finite and infinite iterations by their concurrent versions, define exactly the family of recognizable languages of infinite traces. We get in this way a generalization of Kleene's theorem to languages of infinite traces. We also show that if T* is recognizable, so is T^ω. This result is especially interesting because it means that any possible sufficient condition ensuring the recognizability of T* ensures also the recognizability of T^ω. It should be noted that our result is by no means a direct extension of the results on finite traces. Our proofs of the ω-analog of Ochmanski [Och85] and Métivier [Met86] results make use of drastically different methods since the proof techniques used previously for finite traces do not extend to the case of infinite traces.

In the second section of this paper, we present all the results on finite and infinite traces that we will use in the sequel. Except for the connected components of an infinite trace, they are not new and therefore their proofs are not repeated here. The third section contains our main result on a Kleene theorem for languages of infinite traces. We conclude the paper with a quick presentation of possible further research directions.

2. Preliminaries

We recall in this section the definitions and results used in the sequel. If more details are needed, we refer the reader to [Eil74], [Per84], [HR86] and [PP91] for the theory of infinite words, to [Maz86], [AR88], [Per89] and [Die90] for recent overviews about finite traces and to [Gas90], [GR91] and [Gas91] as regards infinite traces.

2.1. Words

We consider a finite alphabet A and as usual we denote by A^* the free monoid over A, that is, the set of finite words over A. The set of infinite words is $A^{\mathbb{N}}$, also denoted A^{ω}. We set $A^{\infty} = A^* \cup A^{\omega}$ the infinitary free monoid with the usual concatenation. As for finite words, a prefix order may be derived from this concatenation: $u < v$ iff there exists w in A^{∞} such that $v = uw$. For u in A^{∞}, alph(u) denotes the set of letters of u, $|u|$ denotes the length of u and $|u|_a$ denotes the number of occurrences of the letter a in the word u. In the infinitary monoid we can consider the infinite concatenation $u_0u_1u_2\ldots$ of a sequence of words $(u_n)_{n \in \mathbb{N}}$. This operation is classicaly extended to languages. The infinite concatenation of the languages $L_0, L_1, L_2, \ldots$ is denoted by $L_0L_1L_2\ldots$ As usual the infinite concatenation LLL... is simply denoted L^{ω}.
For any subset B of A, Π_B denotes the projection from A^{∞} onto B^{∞} which erases non-members of B. Note that Π_B is not a morphism if it is applied to infinite words.

2.2. Traces

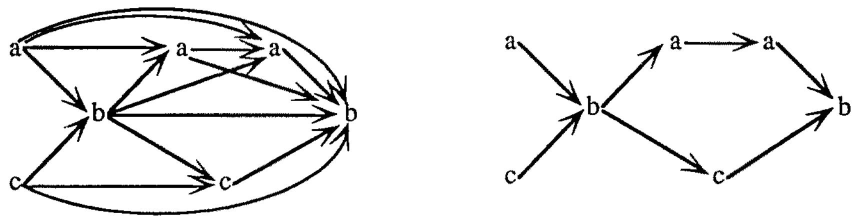
We consider a binary irreflexive and symmetric relation I over the alphabet A called the **independence relation.** The letters of A can be viewed as actions in a distributed system and two actions are independent iff they are related by I. The **dependence relation** over A, denoted by D, is the complement of I: $D = A \times A \setminus I$. According to the relation I, we define an **equivalence relation** $\sim_I$, or simply $\sim$, on sequences of events which is the reflexive and transitive closure of the relation $\{(uabv,ubav) / u,v \in A^*, (a,b) \in I\}$. The relation $\sim$ is a congruence over A^*. The quotient monoid $A^*/\sim$ is called the **free partially commutative monoid** induced by I and is denoted by $M(A^*,I)$. The members of $M(A^*,I)$ are called **traces** and φ will denote the canonical morphism from A^* onto $M(A^*,I)$. The set $A^+/\sim$ of non empty finite traces is denoted by $M(A^+,I)$.
It is possible but not quite natural to extend directly the equivalence relation to infinite words and to define infinite traces as equivalence classes of words. It is much more natural to introduce infinite traces using dependence graphs, which are known as fundamental representations for traces [Maz86], [AR88], [Gas90].

A **dependence graph** is (an isomorphism class of) a labelled acyclic graph (V,E,λ) with V a countable set of vertices, $E \subset V \times V$ a set of edges and $\lambda: V \to A$ a labelling function which satisfies:

(γ_1) $\forall\, x,y \in V,\ (\lambda(x),\lambda(y)) \in D \Leftrightarrow x = y$ or $(x,y) \in E$ or $(y,x) \in E$

(γ_2) $\forall\, x \in V,\ \{y \in V \,/\, (y,x) \in E^*\}$ is finite (where E^* denotes the reflexive and transitive closure of the relation E).

The set of dependence graphs will be denoted $\mathcal{G}(A,D)$. For instance, let $A = \{a,b,c\}$ and $I = \{(a,c),(c,a)\}$, a dependence graph is presented in Figure 2.1(a). Since we are only interested in the isomorphism class of the graph, we only write the labelling of vertices. In the following, in order to lighten the pictures, we will only draw the **Hasse diagram** of the graph, which is the minimal representation of the causal relation E^* on V. For instance, the Hasse diagram associated with the previous dependence graph is presented in Figure 2.1(b).



(a) Figure 2.1 (b)

We define now a **mapping** Γ which associates a dependence graph with each word. Let u be a word in A^∞. The graph $\Gamma(u) = (V,E,\lambda)$ is defined by:

$V = \{\ (a,j) \,/\, a \in A \text{ and } 1 \le j \le |u|_a\ \}$

$E = \{\ (a,j) \to (b,k) \,/\, (a,b) \in D \text{ and the } j^{th}\ a \text{ occurs before the } k^{th}\ b \text{ in } u\ \}$

$\lambda(a,j) = a$ for all (a,j) in V.

For instance, let $A = \{a,b,c\}$, $I = \{(a,c),(c,a)\}$ and $u = acbaacb$. The graph associated with u is presented in Figure 2.2. Note that this is the same graph as in Figure 2.1(a).

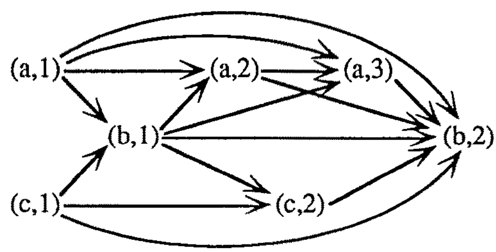


Figure 2.2

The mapping Γ allows us to identify finite traces and finite dependence graphs. Namely, Γ is a surjective mapping from A^* onto $\mathcal{FG}(A,D)$, the set of finite dependence graphs. Moreover, two finite words u and v are equivalent iff $\Gamma(u) = \Gamma(v)$. Therefore, Γ is a bijection between $A^*/\!\sim$ and $\mathcal{FG}(A,D)$. Naturally, we define the **equivalence relation to infinite words** as follows: for all u,v in A^∞, $u \sim v$ iff $\Gamma(u) = \Gamma(v)$. Thus, it turns out that Γ is a bijection between $A^\infty/\!\sim$ and $\mathcal{G}(A,D)$. As in the finite case, φ will denote the canonical mapping from A^∞ onto $A^\infty/\!\sim$. In the following, we will **identify** the equivalence class $\varphi(u)$ of a word u and its dependence graph $\Gamma(u)$.

We present now the **concatenation** on dependence graphs. Let $G_1 = (V_1,E_1,\lambda_1)$ and $G_2 = (V_2,E_2,\lambda_2)$ be two dependence graphs with $V_1 \cap V_2 = \emptyset$. We define $G_1 \otimes G_2$ by:

$$G_1 \otimes G_2 = (V_1 \cup V_2 , E_1 \cup E_2 \cup \{(x,y) \in V_1 \times V_2 / (\lambda_1(x), \lambda_2(y)) \in D\} , \lambda_1 \cup \lambda_2)$$

For instance, Figure 2.3 presents the graphs $\Gamma(acbaacb) \otimes \Gamma(aaccbca^{\omega})$, $\Gamma(aaccbca^{\omega}) \otimes \Gamma(cc)$ and $\Gamma(aaccbca^{\omega}) \otimes \Gamma(b)$.

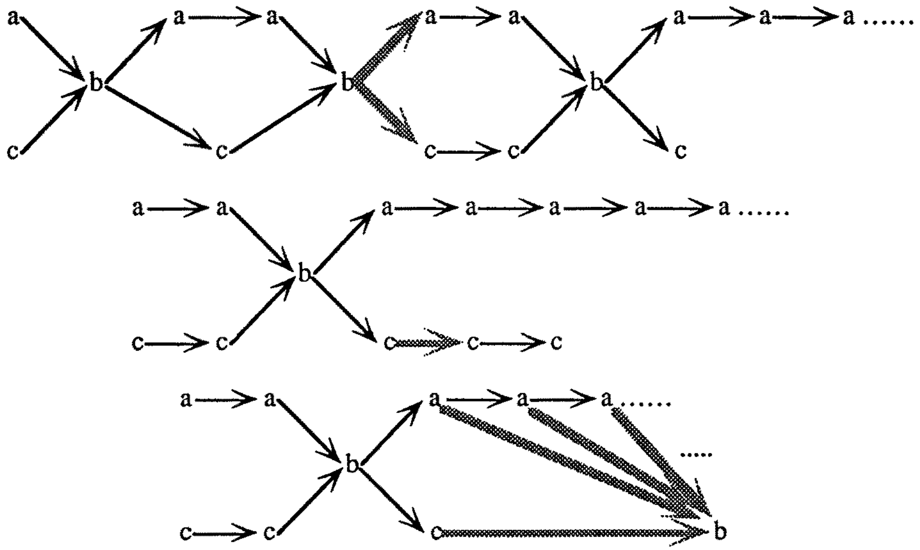


Figure 2.3

Clearly, $G_1 \otimes G_2$ is a labelled acyclic graph with a countable set of vertices and satisfies (γ_1). Now it verifies (γ_2) iff $\mathrm{alphinf}(G_1) \times \mathrm{alph}(G_2) \subset I$, where $\mathrm{alph}(G) = \lambda(V)$ and $\mathrm{alphinf}(G) = \{a \in A / |\lambda^{-1}(a)| = \infty\}$. For instance, the first two graphs of Figure 2.3 satisfy (γ_2), whereas the third one does not.

We introduce the set $\mathcal{G}\mathcal{G}(A,D)$ of labelled acyclic graphs which satisfy (γ_1). These **generalized dependence graphs** do not necessarily satisfy (γ_2). It is easy to see that $(\mathcal{G}\mathcal{G}(A,D),\otimes)$ is a monoid. Let $\mathcal{J}$ be the set of generalized dependence graphs which *do not* satisfy (γ_2). The set $\mathcal{J}$ is clearly an **ideal** of $\mathcal{G}\mathcal{G}(A,D)$. Thus, we define the monoid of **finite and infinite traces** $M(A^{\infty},I)$ as the **Rees quotient** of $\mathcal{G}\mathcal{G}(A,D)$ by the ideal $\mathcal{J}$. We denote by $\perp$ the zero associated with the ideal $\mathcal{J}$ in the Rees quotient.

In other words, the monoid of finite and infinite traces is $M(A^{\infty},I) = \mathcal{G}(A,D) \cup \{\perp\}$. The concatenation on $M(A^{\infty},I)$ is defined as follows: for all G_1, G_2 in $\mathcal{G}(A,D)$, $G_1.G_2 = G_1 \otimes G_2$ if $\mathrm{alphinf}(G_1) \times \mathrm{alph}(G_2) \subset I$ and $G_1.G_2 = \perp$ otherwise. Moreover, $\perp$ is a zero for the concatenation. The element $\perp$ is called **error** because it is used to denote all graphs which do not satisfy (γ_2). When we want to precise that a trace is different from error, we will call it a

real trace. The monoid $M(A^\infty,I)\setminus M(A^*,I)$ of infinite traces is denoted by $M(A^\omega,I)$. Note that $\perp$ is naturally considered as an infinite trace.

The **prefix order** on traces is again defined using dependence graphs. Let G_1, G_2 be in $\mathcal{G}(A,D)$. G_1 is a prefix of G_2, denoted by $G_1 < G_2$, iff G_1 is a subgraph of G_2 (that is, $V_1 \subset V_2$, $E_1 = E_{2\,|\,V_1 \times V_1}$ and $\lambda_1 = \lambda_{2\,|\,V_1}$) which is backward closed (that is, $y \in V_1$ and $(x,y) \in E_2$ imply $x \in V_1$). Moreover, we complete the definition of the prefix order by setting that $\perp$ is the greatest element in $M(A^\infty,I)$.

The following properties [Gas90] are helpfull when one deal with traces. They state that the monoid $M(A^\infty,I)$ is nearly cancellative and that the concatenation is related to the prefix order and to the concatenation on words.

i) $\forall\, u \in A^*, \forall\, v \in A^\infty, \varphi(u).\varphi(v) = \varphi(u.v)$

ii) $\forall\, r,s \in M(A^\infty,I)\setminus\{\perp\}, r < s \Leftrightarrow \exists\, t \in M(A^\infty,I)\setminus\{\perp\}$ such that $s = r.t$

iii) $\forall\, r,s,t \in M(A^\infty,I), r.s = r.t \neq \perp \Rightarrow s = t$

iv) $\forall\, r \in M(A^*,I), \forall\, s,t \in M(A^\infty,I), s.r = t.r \neq \perp \Rightarrow s = t$

For finite traces, a well known characterization of the equivalence relation on words uses **projections on cliques** [CP85]. The generalization to infinite traces of this characterization is the following [Gas90]. Let $\mathcal{C}$ be a covering of A by cliques of the graph (A,D) and let u,v be in A^∞. Then, $u \sim v$ if and only if for each clique C in $\mathcal{C}$, we have $\Pi_C(u) = \Pi_C(v)$. This characterization will be used in proofs of section 3.

The properties of the Poset $(M(A^\infty,I),<)$ are studied in [GR91]. In particular, it is proved that any increasing sequence in $M(A^\infty,I)$ admits a least upper bound in $M(A^\infty,I)$. This property allows us to define the concatenation of an infinite sequence of traces. Let $(s_i)_{i \in \mathbb{N}}$ be a sequence of traces, the **infinite concatenation** of this sequence is the least upper bound of its partial products, that is, $s_0s_1s_2\ldots = \sqcup\{s_0s_1\ldots s_n, n \in \mathbb{N}\}$. Note that either the partial products are ultimately equal to $\perp$ and the infinite product is equal to $\perp$, or the partial products form an increasing sequence of real traces and the infinite product is a real trace. Note also that as soon as we use more than $|A|$ infinite traces, the infinite product is inevitably equal to $\perp$. Therefore, an infinite product which is not error uses only a finite number of infinite traces. In the restricted case of finite traces we can relate the infinite product of traces to the infinite product of words. More precisely, let $(u_i)_{i \in \mathbb{N}}$ be a sequence of finite words. Then it is easy to verify that $\varphi(u_0)\varphi(u_1)\varphi(u_2)\ldots = \varphi(u_0u_1u_2\ldots)$. We are now able to define the **infinite concatenation** of trace languages and therefore the **infinite iteration** of a trace language.

i) $\forall\, i \in \mathbb{N}$, let $T_i \subset M(A^\infty,I)$. We set $T_0.T_1.T_2\ldots = \{t_0t_1t_2\ldots \,/\, \forall\, i \in \mathbb{N}, t_i \in T_i\}$

ii) Let $T \subset M(A^\infty,I)$, we set $T^\omega = (T\setminus\{\varepsilon\}).(T\setminus\{\varepsilon\}).(T\setminus\{\varepsilon\})\ldots$

2.3. Recognizable languages

In the finitary partially commutative monoid the definition of recognizable languages is classical: $T \subset M(A^*,I)$ is **recognizable** iff there exists a morphism η from $M(A^*,I)$ into a finite monoid which recognizes T (i.e. $\eta^{-1}(\eta(T)) = T$). When we deal with infinite traces the definition of a recognizable language is similar. A morphism η from $M(A^+,I)$ into a finite semigroup S **recognizes** a language T of $M(A^\omega,I)$ iff

$\forall\ (t_n)_{n\in N} \subset M(A^+,I),\ t_0t_1t_2\ldots \in T \Rightarrow \eta^{-1}(\eta(t_0)).\eta^{-1}(\eta(t_1)).\eta^{-1}(\eta(t_2))\ldots \subset T.$

Note that η recognizes T iff η recognizes $T\setminus\{\perp\}$. In this case it was proved in [Gas91] that

$$T\setminus\{\perp\} = \bigcup_{(s,e)\in P} \eta^{-1}(s).\eta^{-1}(e)^{\omega} \text{ with } P = \{(s,e) \in S^2 \,/\, se=s,\ e^2=e \text{ and } \eta^{-1}(s).\eta^{-1}(e)^{\omega} \subset T\}.$$

Finally a trace language $T \subset M(A^{\infty},I)$ is **recognizable** iff $T = T_{fin} \cup T_{inf}$ where T_{fin} and T_{inf} are recognizable languages of finite and infinite traces respectively.

We denote by $Rec(A^*,I)$, $Rec(A^{\omega},I)$, and $Rec(A^{\infty},I)$ the sets of recognizable languages of $M(A^*,I)$, $M(A^{\omega},I)$ and $M(A^{\infty},I)$ respectively. In the case of the empty independence relation, we obtain the classical definitions of recognizable families of word languages, which are simply denoted by $Rec(A^*)$, $Rec(A^{\omega})$, and $Rec(A^{\infty})$.

In fact, recognizable trace languages and recognizable word languages are closely related by the following properties:

$T \in Rec(A^*,I)$ iff $\varphi^{-1}(T) \in Rec(A^*)$.

$T \in Rec(A^{\omega},I)$ iff $\varphi^{-1}(T) \in Rec(A^{\omega})$ [Gas90].

In consequence, well-known closure properties of $Rec(A^*)$, $Rec(A^{\omega})$ and $Rec(A^{\infty})$ imply that the three families $Rec(A^*,I)$, $Rec(A^{\omega},I)$ and $Rec(A^{\infty},I)$ are closed under the Boolean operations union, intersection and complement.

2.4. Rational languages

The family of **rational** trace languages of $M(A^*,I)$ denoted by $Rat(A^*,I)$ is the least family which contains the empty set $\emptyset$, the languages $\{a\}$ for each letter a and which is closed under union, concatenation and iteration $(*)$. The definition is similar for the family $Rat(A^{\infty},I)$ of rational trace languages of $M(A^{\infty},I)$ but requires in addition the closure by the infinite iteration (ω). The classical definitions of rational word languages $Rat(A^*)$ and $Rat(A^{\infty})$ are similar. Note that, since the concatenation in $M(A^{\infty},\emptyset)$ does not coincide with the classical concatenation in the free monoid A^{∞}, rational expressions have different interpretations in $Rat(A^{\infty},\emptyset)$ and $Rat(A^{\infty})$. More precisely, the mapping $T \to T\setminus\{\perp\}$ from $Rat(A^{\infty},\emptyset)$ in $Rat(A^{\infty})$ is not a morphism for rational operations. Nevertheless, we have $T \in Rat(A^{\infty},\emptyset)$ iff $T\setminus\{\perp\} \in Rat(A^{\infty})$.

Kleene's theorem states that a word language is recognizable iff it is rational. This is not true for trace languages where it holds only that any recognizable language is rational. The problem comes from the iteration. For instance, let $A = \{a,b,c\}$ and $I = \{(a,c),(c,a)\}$ then we get $\{ac\}^* \notin Rec(A^*,I)$. It is a little bit surprising that the same example doesn't work for the infinite iteration $(\{ac\}^{\omega} \in Rec(A^{\omega},I))$. In fact, for any finite trace t, $\{t\}^{\omega} \in Rec(A^{\omega},I)$ [Gas91]. But in general the family $Rec(A^{\infty},I)$ is not closed by infinite iteration since $\{ac,b\}^{\omega} \notin Rec(A^{\omega},I)$.

Some closure properties were nevertheless proved for the families of recognizable trace languages. We have already stated the closure by the Boolean operations. The closure by concatenation was proved independently in [Fli74] and [CP85] for finite traces and in [Gas91] for infinite traces.

Many researches were done in order to find sufficient conditions which ensure that the iteration of a recognizable language is still recognizable. One of them is the connectivity of the trace language. A real trace t is **connected** iff its dependence graph $\Gamma(t)$ is connected. The property of connectivity for a trace can be defined, in an equivalent way, using the alphabet of the trace. A subset B of A is connected iff the restriction of the dependence graph D to $B \times B$ is connected, a real trace t is connected iff alph(t) is connected. A trace language T is connected iff all traces in T are connected. An important result [Met86], [Och85] states that if a recognizable language $T \in Rec(A^*,I)$ is connected then T^* is recognizable too.

2.5. Co-rational languages

Now we give the definition of the **connected components** of a real trace. Let t be a real trace in $M(A^\infty,I) \setminus \{\perp\}$. The connected components of t are the traces associated with the connected components of the dependence graph $\Gamma(t)$. We denote by C(t) the set of connected components of t. Clearly, t is the commutative product of its connected components, that is, $t = t_1t_2\ldots t_k$ with $C(t) = \{t_1,\ldots,t_k\}$. Moreover, we set $C(\perp) = \{\perp\}$. For a trace language T, we set $C(T) = \bigcup_{t \in T} C(t)$. An important property of connected components is that if T is in $Rec(A^\infty,I)$ then C(T) is recognizable too. This generalizes the same result for finite traces [Och85], the proof is left to the reader.

The **concurrent iteration** of a trace language T is simply $T^{co\text{-}*} = C(T)^*$. We extend this definition to the concurrent infinite iteration $T^{co\text{-}\omega} = C(T)^\omega$. The families of **co-rational** languages $co\text{-}Rat(A^*,I)$ and $co\text{-}Rat(A^\infty,I)$ are defined like the rational families but use the concurrent iterations co-$*$ and co-ω instead of the iterations $*$ and ω.

One important result in trace theory is the Kleene's theorem for finite traces, due to Ochmanski [Och85], which asserts the equality between the families $Rec(A^*,I)$ and $co\text{-}Rat(A^*,I)$. As we explained in the introduction the aim of this paper is to extend this result to infinite traces.

3. A Kleene theorem

We have already seen that the set $Rec(A^\infty,I)$ is not closed under infinite iteration. If $I = \{(a,c),(c,a)\}$ then $T = \{ac,b\}$ is recognizable whereas $T^\omega = \{ac,b\}^\omega$ is not. First, we will study sufficient conditions ensuring the recognizability of T^ω for a trace language T of $Rec(A^*,I)$. In the previous example the elements of T have different alphabets. Therefore we can wonder whether T^ω remains recognizable when T is a recognizable language and all traces of T have the same alphabet. Unfortunately, this is not the case: if $I = \{(a,c), (c,a), (a,d), (d,a), (b,c), (c,b), (b,d), (d,b)\}$ and $T = \{abcd, abbcdd\}$ the set T^ω is not recognizable since $\varphi^{-1}(T^\omega) \cap \{ab\}^*\{cd\}^*\{abbcdd\}^\omega = \{(ab)^n(cd)^n(abbcdd)^\omega / n \geq 0\}$. Nevertheless, if we impose that the common alphabet of all the traces of T is connected then the set T^ω is still recognizable.

Proposition 3.1. Let B be a connected subset of A and let T be a language of $Rec(A^*,I)$ such that $alph(t) = B$ for any t in T. Then T^ω is a language of $Rec(A^\omega,I)$.

Proof. (Sketch) According to Section 2.3 it is sufficient to prove that $H = \varphi^{-1}(T^\omega)$ is a recognizable language of A^ω.

Let us denote $L = \varphi^{-1}(T)$, by hypothesis L is a recognizable language of A^*. Let us first give an intuition of the proof using an infinite number of pairwise disjoint copies of A: $A_1, A_2, \ldots$

The copy in A_i of a letter a of A is a_i. We denote by $\tilde{A}$ the union $\bigcup_{i\geq 1} A_i$. Let h be the morphism from $\tilde{A}$ onto A defined by $h(a_i) = a$, and let h_i be the morphism from A onto A_i defined by $h_i(a) = a_i$. Let $L_i = h_i(L)$. From the recognizable languages L_i, we define a language of $\tilde{A}^{\omega}$: $M' = \underset{i\geq 1}{ш} L_i = \{u \in \tilde{A}^{\omega} / \forall\ i \geq 1, h_i(u) \in L_i\}$ (where ш denotes the shuffle operation).

In order to obtain the language H from M' (through the morphism h), we must remove all the words where a letter b_j precedes a letter a_i when $j>i$ and $(h(a_i),h(b_j)) \in D$. Formally we define the set R' by its complementary $(R')^c = \bigcup_{j>i\ \&\ (a,b)\in D} \tilde{A}^* b_j \tilde{A}^* a_i \tilde{A}^{\omega}$.

It is then quite easy to show that $h(M' \cap R') = H$. If $\tilde{A}$ were a finite alphabet, the sets M' and R' would be recognizable . Therefore the proposition would be proved.

The possibility to replace $\tilde{A}$ by a finite alphabet is provided by the fact that the common alphabet of the elements of T is connected. More precisely we use the following result. Let k be the size of the alphabet B. We denote by I' the independence relation defined by $(a_i,b_j) \in I'$ iff $(h(a_i),h(b_j)) \in I$. For any integer k, [k] denotes the set $\{1,\ldots,k\}$ and $i = j\ [k]$ means that i equals j modulo k. For i in [k], let Π_i be the projection from $\tilde{A}^{\infty}$ onto $\left(\bigcup_{j=i\ [k]} A_j\right)^{\infty}$.

Fact: Let (u_i) be an infinite sequence of words which verify $alph(u_i) = h_i(B)$. Then, in any word v equivalent to $u_1u_2\ldots u_k\ldots$, we have:

$\Pi_i(v) \in [u_i]_{I'}[u_{i+k}]_{I'}[u_{i+2k}]_{I'}\ldots$ (where $[w]_{I'}$ denotes the set of words $\sim_{I'}$ equivalent to w).

This result is a direct extension to infinite traces of a similar one on finite connected traces used for instance in [CM85], [Met86] or [CL87]. The proof is thus left to the reader.

Therefore, we can define new recognizable languages M and R using only 2k colours. We denote by $\hat{A}$ the union $A_1\cup\ldots\cup A_{2k}$ and we define a set M, the analogue of M':

$$M = \underset{1\leq i\leq k}{ш} (L_iL_{i+k})^{\omega} = \{\ u \in \hat{A}^{\omega} / \forall\ i \in [k], \Pi_i(u) \in (L_iL_{i+k})^{\omega}\ \}.$$

From classical results on recognizable languages, M is recognizable. To define the recognizable language R, the analogue of R', we need to introduce a subset X of couples (a_i,b_j).

$$X = \{\ (a_i,b_j) / (h(a_i),h(b_j)) \in D,\ 1 \leq i,j \leq 2k \text{ and } i < j < i+k \text{ or } j < i-k\}$$

The language R is defined by its complementary $R^c = \bigcup_{(a_i,b_j)\in X} \hat{A}^* b_j (\hat{A} \setminus \{a_{i+k\ [2k]}\})^* a_i \hat{A}^{\omega}$.

At last, we show that $H = h(M \cap R)$ and thus the language H is recognizable. ◊

It is already known that under the hypotheses of Proposition 3.1, T^* is a recognizable trace language [Met86], [Och85]. In fact, one of our main result states that as soon as T^* is a recognizable trace language, so is T^{ω}. Before proving this result we give two lemmas on the structure of languages of the form T^{ω}.

Lemma 3.2. Let T be a language in $M(A^*,I)$ such that $T^+ = T$. Then,

$$T^{\omega} = \bigcup_{B\subset A} TT_B^{\omega} \text{ where } T_B = \{\ t \in T / alph(t) = B\}.$$

Proof. Easy. ◊

Lemma 3.3. Let B be a subset of A, let $B_1,\ldots,B_k$ be the connected components of B and let T be a language of $M(A^*,I)$ such that T^+ is in $Rec(A^*,I)$ and alph(t) = B for all t in T. Then T^ω is a finite union of sets of the form $XY_1^\omega\ldots Y_k^\omega$ where $X,Y_1,\ldots,Y_k$ are in $Rec(A^*,I)$ and $alph(t) = B_i$ for all t in Y_i.

Proof. (Sketch) Since $T^\omega = (T^+)^\omega$, w.l.o.g. we assume that $T = T^+$. Using Mezei's theorem [Eil74], we obtain $T = \bigcup_{i\in[n]} T_i$ for some integer n with $T_i = T_{1,i}\ldots T_{k,i}$, $T_{j,i} \in Rec(A^*,I)$ and $alph(t) = B_j$ for all t in $T_{j,i}$. We claim that $T^\omega = \bigcup_{i,j\in[n]} T_iT_j^\omega$. One inclusion is trivial. The opposite inclusion can be proved using Ramsey's factorizations [PP91]. Lemma 3.3 follows then directly from the claim since $T_j^\omega = T_{1,j}^\omega\ldots T_{k,j}^\omega$. ◊

We are now ready to prove one of our main results.

Theorem 3.5. Let T be a language of $M(A^*,I)$ such that T^* is in $Rec(A^*,I)$. Then T^ω is in $Rec(A^\omega,I)$.

Proof. Since $T^\omega = (T^+)^\omega$, w.l.o.g. we assume that $T = T^+$. From Lemma 3.2 we obtain $T^\omega = \bigcup_{B\subset A} TT_B^\omega$ with $T_B = \{\, t \in T \,/\, alph(t) = B\}$. Let B be a subset of A. Clearly, $T_B^+ = T_B$ is in $Rec(A^*,I)$ and we can apply Lemma 3.3. Let $B_1,\ldots,B_k$ be the connected components of B. T_B^ω is a finite union of sets of the form $XY_1^\omega\ldots Y_k^\omega$ where $X,Y_1,\ldots,Y_k$ are in $Rec(A^*,I)$ and $alph(t) = B_i$ for all t in Y_i. By Proposition 3.1, the languages Y_i^ω are recognizable. The result follows then from the closure properties of $Rec(A^\omega,I)$ recalled in Section 2.4. ◊

An important result on finite traces theory [Met86], [Och85] states that if T is in $Rec(A^*,I)$ and if all the traces of T are connected then T^* is also in $Rec(A^*,I)$. Therefore we obtain as corollary of Theorem 3.5 a similar result on T^ω which can be considered also as a generalization of Proposition 3.1.

Corollary 3.6. Let T be a language in $Rec(A^*,I)$ such that all the traces of T are connected. Then T^ω is in $Rec(A^\omega,I)$.

All the preliminaries concerning sufficient conditions for the recognizability of a language T^ω being done, we can now prove that the set of recognizable languages is closed under co-rational operations and that, conversely, any recognizable language is co-rational.

Theorem 3.7. $co\text{-}Rat(A^\infty,I) = Rec(A^\infty,I)$

Proof. First, we prove that every co-rational language is recognizable. Since $Rec(A^\infty,I)$ is closed under boolean operations, concatenation and decomposition in connected components (Sections 2.3, 2.4 and 2.5), it remains to prove that if T is a language of $Rec(A^\infty,I)$ which consists of connected traces only then T^* and T^ω are recognizable too. Since a product which contains more than $|A|$ infinite traces is equal to error, we have $T^* = \bigcup_{0\le i\le |A|} (T_{fin}{}^* T_{inf})^i T_{fin}{}^*$

and $T^\omega = T^* T_{fin}{}^\omega$ with $T_{fin} = T \cap M(A^*,I)$ and $T_{inf} = T \cap M(A^\omega,I)$. The result follows since $T_{fin}{}^*$ is recognizable [Met86], [Och85] and from Corollary 3.6 $T_{fin}{}^\omega$ is also recognizable.

Thus it remains to prove the converse inclusion. Let $T = T_{fin} \cup T_{inf}$ be in $Rec(A^\infty,I)$. There exists (Section 2.3) a morphism η from $M(A^+,I)$ into a finite semigroup S such that:

$$T_{inf} \setminus \{\perp\} = \bigcup_{(s,e) \in X} \eta^{-1}(s)(\eta^{-1}(e))^\omega \text{ with } X = \{(s,e) \in S^2 \mid se=s,\ e^2=e,\ \eta^{-1}(s)(\eta^{-1}(e))^\omega \subset T_{inf}\}.$$

Set $R = \eta^{-1}(e)$ for some e in S such that $e^2 = e$. The last equality implies that $R^+ = R$ and then from Lemma 3.2 we have $R^\omega = \bigcup_{B \subset A} RR_B{}^\omega$. Let B be a subset of A and let $B_1,\ldots,B_k$ be the connected components of B. Since $R_B{}^+ = R_B$, by Lemma 3.3, $R_B{}^\omega$ is a finite union of sets of the form $XY_1{}^\omega \ldots Y_k{}^\omega$ where $X,Y_1,\ldots,Y_k$ are in $Rec(A^*,I)$ and $alph(t) = B_i$ for all t in Y_i. Since the sets Y_i consists of connected traces only, we have $Y_i{}^\omega = Y_i{}^{co\text{-}\omega}$. Then, using Ochmanski's theorem $Rec(A^*,I) = co\text{-}Rat(A^*,I)$ [Och85], we deduce that T is co-rational. ◊

It is well known that a language L of A^ω is recognizable if and only if L is a finite union of languages of the form XY^ω where X and Y are recognizable languages of A^*. The proof of Theorem 3.7 allows to exhibite a comparable canonical form for the languages of $Rec(A^\omega,I)$.

Theorem 3.8: A language T is in $Rec(A^\omega,I)$ iff $T \setminus \{\perp\}$ is a finite union of languages of the form $XY_1{}^\omega \ldots Y_k{}^\omega$ where X and Y_i are in $Rec(A^*,I)$ and there exist connected subsets $B_1,\ldots,B_k$ of A which verify $B_i \times B_j \subset I$, for $i \neq j$ and $alph(t) = B_i$ for all t in Y_i.

4. Conclusion

In this paper we proposed a Kleene theorem for languages of finite and infinite traces. We have obtained also a canonical rational form for recognizable trace languages.

Both for recognizable languages of infinite words and for recognizable languages of finite traces, there exist also characterizations by automata (Büchi [Büc62] or Muller [Mul63] automata for infinite words and diamond automata, asynchronous [Zie87] or cellular asynchronous [CMZ90] automata for finite traces). There exist also characterization of recognizable languages by logical formulae (see e.g. [Büc62] or [Tho88] for infinite words and [Tho89] for finite traces). The problem of finding similar characterizations for infinite traces remains open.

Our main theorem and Ochmanski's theorem prove that the good notion of iteration for traces is the so-called concurrent iteration. Note that when the independence relation is empty, this concurrent iteration reduces to the usual Kleene's iteration.

Acknowledgements: *The authors thank V. Diekert, J.E. Pin and J. Sakarovitch for many helpful comments and advices.*

5. References

[AR88] I.J. AALBERSBERG and G. ROZENBERG, "Theory of traces", Theoretical Computer Science 60, p. 1-82, 1988.

[BMP89] P. BONIZZONI, G. MAURI and G. PIGHIZZINI, "About infinite traces", Proceedings of the ASMICS Workshop on Partially Commutative Monoids, Tech. Rep. TUM-I 9002, Technische Universität München, 1989.

[Büc62] J.R. BUCHI, "On a decision method in restricted second order arithmetic", Proc. Internat. Congress on Logic, Methodology and Philosophy (Standford University Press), p. 1-11, 1962.

[CF69] P. CARTIER and D. FOATA, "Problèmes combinatoires de commutation et réarrangements", Lecture Notes in Math. 85, 1969.

[CL87] M. CLERBOUT and M. LATTEUX, "Semi-commutations", Information and Computation 73, 1987.

[CM85] R. CORI and Y. METIVIER, "Recognizable subsets of some partially abelian monoids", Theoretical Computer Science 35, p. 179-189, 1985.

[CMZ89] R. CORI, Y. METIVIER and W. ZIELONKA, "Asynchronous mappings and asynchronous cellular automata", Tech. Rep. 89-97, LaBRI, Université de Bordeaux, France, 1990.

[CP85] R. CORI and D. PERRIN, "Automates et commutations partielles", RAIRO Theoretical Informatics and Applications 19, p. 21-32, 1985.

[Die90] V. DIEKERT, "Combinatorics on traces", Lecture Notes in Computer Science 454, 1990.

[Die91] V. DIEKERT, "On the concatenation of infinite traces", STACS'91, Lecture Notes in Computer Science 480, p. 105-117, 1991.

[Dub86] C. DUBOC, "Commutations dans les monoïdes libres: un cadre théorique pour l'étude du parallélisme", Thèse, Université de Rouen, France, 1986.

[Eil74] S. EILENBERG, "Automata, Languages and Machines", Academic Press, New York, 1974.

[Fli74] M. FLIESS, "Matrices de Hankel", J. Math. pures et appl. 53, p. 197-224, 1974.

[FR85] M.P. FLE and G. ROUCAIROL, "Maximal serializability of iterated transactions", Theoretical Computer Science 38, p. 1-16, 1985.

[Gas88] P. GASTIN, "Un modèle asynchrone pour les systèmes distribués", Theoretical Computer Science 74, p. 121-162, 1990.

[Gas90] P. GASTIN, "Infinite traces", Proceedings of the Spring School of Theoretical Computer Science on "Semantics of concurrency", Lecture Notes in Computer Science 469, p. 277-308, 1990.

[Gas91] P. GASTIN, "Recognizable and rational languages of finite and infinite traces", STACS'91, Lecture Notes in Computer Science 480, p. 89-104, 1991.

[Gra81] R. L. GRAHAM, "Rudiments of Ramsey theory ", Regional conference series in mathematics 45, 1981.

[GR91] P. GASTIN and B.ROZOY, "The Poset of infinitary traces", to appear in Theoretical Computer Science, Tech. Rep. 91-07, LITP, Université Paris 6, France, 1991.

[HR86] H.J. HOOGEBOOM and G. ROZENBERG, "Infinitary languages: basic theory and applications to concurrent systems", Lecture Notes in Computer Science 224, p. 266-342, 1986.

[Kwi89] M.Z. KWIATKOWSKA, "On infinitary trace languages", Tech. Rep. 31, University of Leicester, England, 1989.

[Maz77] A. MAZURKIEWICZ, "Concurrent program Schemes and their interpretations", Aarhus University, DAIMI Rep. PB 78, 1977.

[Maz86] A. MAZURKIEWICZ, "Trace theory", Advanced Course on Petri Nets, Lecture Notes in Computer Science 255, p. 279-324, 1986.

[Met86] Y. METIVIER, "Une condition suffisante de reconnaissabilité dans un monoïde partiellement commutatif", RAIRO Theoretical Informatics and Applications 20, p. 121-127, 1986.

[Met86] Y. METIVIER, "On recognizable subsets in free partially commutative monoids", ICALP 1986, Lecture Notes in Computer Science 226, p. 254-264, 1986.

[MR91] Y. METIVIER and B.ROZOY, "On the star operation in free partially commutative monoids", Tech. Rep. 625, LRI, Université de Paris Sud, France, 1991.

[Mul63] D.E. MULLER, "Infinite sequences and finite machines", Proc. 4th IEEE Ann. Symp. on Switching Circuit Theory and Logical Design, p. 3-16, 1963.

[Och85] E. OCHMANSKI, "Regular behaviour of concurrent systems", Bulletin of EATCS 27, p. 56-67, October 1985.

[Och90] E. OCHMANSKI, "Notes on a star mystery", Bulletin of EATCS 40, p. 252-257, February 1990.

[Per84] D. PERRIN, "Recent results on automata and infinite words", Lecture Notes in Computer Science 176, p. 134-148, 1984.

[Per89] D. PERRIN, "Partial commutations", ICALP 89, Lecture Notes in Computer Science 372, p. 637-651, 1989.

[PP91] D.PERRIN and J.E. PIN, "Mots Infinis", Book to appear, Tech. Rep. 91-06, LITP, Université Paris 6, France, 1991.

[Roz90] B. ROZOY, "On Traces, Partial Order Sets and Recognizability", ISCIS V, Cappadocia, Turkey, proceedings to appear, 1990.

[Sak87] J. SAKAROVITCH, "On regular trace languages", Theoretical Computer Science 52, p. 59-75, 1987.

[Tho88] W. THOMAS, "Automata on infinite objects", to appear in Handbook of Theoretical Computer Science (J.V. Leeuwen, Ed.), North-Holland, Amsterdam.

[Tho89] W. THOMAS, "On logical definability of trace languages", Proceedings of the ASMICS Workshop on Partially Commutative Monoids, Tech. Rep. TUM-I 9002, Technische Universität München, 1989.

[Zie87] W. ZIELONKA, "Notes on finite asynchronous automata and trace languages", RAIRO Theoretical Informatics and Applications 21, p. 99-135, 1987.