Transitive Closures of Semi-commutation Relations on Regular omega-Languages

Pierre-Cyrille Héam

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Transitive Closures of Semi-commutation Relations on Regular ω-Languages

P.-C. Héam

N° 6239

June 2007

THÈMES 4 et 3

Rapport de recherche

N 0249-6399

Transitive Closures of Semi-commutation Relations on Regular ω-Languages

P.-C. Héam

Thèmes 4 et 3 — Simulation et optimisation
de systèmes complexes — Interaction homme-machine,
images, données, connaissances
Projets CASSIS

Rapport de recherche n° 6239 — June 2007 — 17 pages

Abstract: A semi-commutation R is a relation on a finite alphabet A. Given an infinite word u on A, we denote by $R(u) = \{xbay \mid x \in A^*, y \in A^\omega\ (a,b) \in R \text{ and } xaby = u\}$ and by $R^*(u)$ the language $\{u\} \cup \cup_{k \geq 1} R^k(u)$. In this paper we prove that if an ω-language L is a finite union of languages of the form $A_0^* a_1 A_1^* \ldots a_k A_k^* a_{k+1} A_{k+1}^*$, where the A_i's are subsets of the alphabet and the a_i's are letters, then $R^*(L)$ is a computable regular ω-language accepting a similar decomposition. In addition we prove the same result holds for ω-languages which are finite unions of languages of the form $L_0 a_1 L_1 \ldots a_k L_k a_{k+1} L_{k+1}$, where the L_i's are accepted by diamond automata and the a_i's are letters. These results improve recent works by Bouajjani, Muscholl and Touili on one hand, and by Cécé, Héam and Mainier on the other hand, by extending them to infinite words.

Key-words: Finite Automata, Infinite Words, Transitive Closures, Semi-commutations

Unité de recherche INRIA Lorraine
LORIA, Technopôle de Nancy-Brabois, Campus scientifique,

Clôtures transitives de realtions de semi-commutation sur les ω-langages réguliers

Résumé : Une relation de semi-commutation R est une relation sur un alphabet fini A. Etant donné un mot infini u sur A, on pose $R(u) = \{xbay \mid x \in A^*, y \in A^\omega\ (a,b) \in R \text{ and } xaby = u\}$ et $R^*(u)$ le langage $\{u\} \cup \cup_{k\geq 1} R^k(u)$. Dans cet article nous montrons que si un ω-langage L est une union finie de langages de la forme $A_0^* a_1 A_1^* \ldots a_k A_k^* a_{k+1} A_{k+1}^*$, où les A_i sont des sous-ensembles de l'alphabet et les a_i des lettres, alors $R^*(L)$ est un ω-langage régulier calculable et possédant une décomposition similaire. De plus, nous prouvons que le même résultat existe pour les ω-langages qui sont une union finie de langages de la forme $L_0 a_1 L_1 \ldots a_k L_k a_{k+1} L_{k+1}$, où les L_i sont acceptés par des automates diamants et les a_i des lettres. Ces résultats étendent aux mots infinis des travaux récents de Bouajjani, Muscholl et Touili d'une part, et Cécé, Héam et Mainier d'autre part.

Mots-clés : Automates finis, mots infinis, clôtures transitives, semi-commutations

We assume a basic background in finite automata theory. For more information on automata the reader is referred to [Ber79, HU80]. We also assume that the reader is familiar with notions on finite/infinite words and languages. For precise definitions the reader could refer to [PP04].

1 Introduction

1.1 Contributions

The main purpose of the paper is to prove stability results on several classes of regular ω-languages. More precisely, we are interested in semi-commutation relations: a semi-commutation R is a relation on a finite alphabet A. Given an infinite word u on A, we denote by $R(u) = \{xbay \mid x \in A^*, y \in A^\omega, (a,b) \in R \text{ and } xaby = u\}$. We denote by $R^*(u)$ the ω-language $\{u\} \cup \cup_{k \geq 1} R^k(u)$. By extension, for an ω-language L, we set

$$R(L) = \cup_{u \in L} R(u) \quad \text{and} \quad R^*(L) = \cup_{u \in L} R^*(u).$$

We say that a finite automaton $\mathcal{A}$ is a diamond automaton[MP01] if for each pair of transitions of the form (p,a,q), (q,b,r), there exists a state s of $\mathcal{A}$ such that (p,b,s) and (s,a,r) are transitions too. Finally, we say that a finite automaton $\mathcal{A}$ is a partially ordered automaton [TT02] if there exists a partial order $\leq$ on its set of states such that for each transition (p,a,q) of $\mathcal{A}$, $p \leq q$.

The main results of this paper are as follows:

(1) We prove that the class of ω-languages accepted by partially ordered Büchi automata is closed under semi-commutation; i.e. if L is accepted by a partially ordered Büchi automaton then, for each semi-commutation relation R, $R^*(L)$ is accepted by a partially ordered Büchi automaton too.

(2) We prove that the class of ω-languages, called $\omega - \mathrm{Pol}\mathcal{C}$, that are finite unions of ω-languages of the form
$$L_0 a_1 L_1 \dots a_k L_k a_{k+1} L_{k+1},$$
where the L_i's are accepted by diamond automata and the a_i's are letters is closed under semi-commutations.

(3) We provide an automaton based algorithm to compute $R^*(L)$ for the two above cases.

In order to obtain this results, we have to use the $\sqcup\!\sqcup_R$ operator. Given two words $u \in A^*$ and $v \in A^* \cup A^\omega$, the R-shuffle of u and v, denoted $u \sqcup\!\sqcup_R v$, is the set of words of the form $u_1 v_1 \dots u_n v_n$ with $u = u_1 \dots u_n$, $v = v_1 \dots v_n$ and such that $\alpha(u_i) \times \alpha(v_j) \subseteq R$ for all $j < i$. The R-shuffle operation is extended to languages $L \subseteq A^*$ and $K \subseteq A^*$ or $K \subseteq A^\omega$ by

$$L \sqcup\!\sqcup_R K = \cup_{u \in L, v \in K} u \sqcup\!\sqcup_R v.$$

In this paper we obtain the following results for the $\sqcup\!\sqcup_R$ operator.

(4) If L is a regular language on finite words and K is a regular ω-language, then $L \sqcup\!\sqcup_R K$ is a regular ω-language.

(5) We provide a polynomial-time algorithm to compute $L \sqcup\!\sqcup_R K$.

(6) We prove that the classes defined in (1) and (2) are closed under the $\sqcup\!\sqcup_R$ operator.

Results (1-6) extend results obtained on finite words by Bouajjani et al. [BMT01, BMT07] and by Cécé et al. [CHM03] to infinite words.

1.2 Related Works

Regular model-checking [BG96, BW98, AJNd03] is an approach to verify infinite state systems. One represents, symbolically, sets of states by regular languages and one develops *meta-transitions* which can compute, in one step, infinite sets of successors. This amounts to compute $R^*(L)$ for a given regular language L and a given relation R representing a subset of the transition relation T of the system. The transition relation T can be decomposed into several (sub) relations R_i (of semi-commutation or something else), each of them implying their ad-hoc techniques of computation. As most of the developed techniques are based on automata, it is more efficient and consistent to use automata during the whole computation. As explained in [BMT07], these techniques also are suitable for verifying of High-level Message Sequence Charts using both finite and infinite executions [GM05]. In this direction our works may have several applications. Moreover, diamond automata play a significant role in the translation of Büchi automata into HMSC's [MP01].

Polynomial closure of varieties of regular languages is an operation widely studied in the literature (see for example [PW97, Tho82, Brz76, BS73]). Languages on finite words accepted by partially ordered automata are called languages of level 3/2 in the Straubing-Thérien hierarchy [Str85, Thé81] which represents the current border for decidability problems and whose structure makes them suitable for verification of certain systems [ABJ98, AAB99, BMT07] [BMT01, Tou01].

Decomposable languages form a class of regular languages used for the simulation of process algebra [LS98]. It was conjectured in [Sch99] that this class was exactly the dual class of $\omega - \mathrm{Pol}\mathcal{C}$ for finite words. However this conjecture has been invalidated in [GP03]. Finally, looking for the maximal (positive) variety closed under an operator [BBC+06] is widely studied in the literature. One can cite the result for the shuffle operator for varieties [ES98, Per78] and for positive varieties [GP04].

The shuffle product is an operation on languages which is strongly connected to combinatorics on words and which was widely studied in the literature [Rad79, Spe86, NRR+94, PMR98, BB99].

1.3 Layout of the paper

After introducing the main issues of this paper and basic notations, we extend in Section 2 a result proved in [DM97] to infinite words and we prove that computing the R-closure of a regular language reduces in some cases to the computation of the R-shuffle of these languages. Then, we provide an algorithm to compute the R-shuffle of two regular languages. Section 3 is dedicated to proving the main contributions of the paper. Finally, we conclude in Section 4 by giving some future works.

1.4 Background and Notations

We recall in this section notations and unusual definitions on words and automata.

Recall that a finite automaton is a 5-tuple $\mathcal{A} = (Q, A, E, I, F)$ where Q is a finite set of states, A is the alphabet, $E \subseteq Q \times A \times Q$ is the set of transitions, $I \subseteq Q$ is the set of initial states and $F \subseteq Q$ is the set of final states. If $\mathcal{A}$ is a finite automaton, $L(\mathcal{A})$ denotes the language accepted by $\mathcal{A}$. If $C \subseteq Q$ and $D \subseteq Q$, $\mathcal{A}_{C,D}$ denotes the automaton (Q, A, E, C, D). Moreover, for all $p \in Q$, $p \cdot_{\mathcal{A}} a = \{q \in Q \mid (p, a, q) \in E\}$. If there is no ambiguity on $\mathcal{A}$, $p \cdot_{\mathcal{A}} a$ is also denoted $p \cdot a$. If $p \cdot a = \{q\}$ is a singleton, we also write $p \cdot a = q$. If $q \in p \cdot a$, we also write $p \rightarrow_a q$.

A finite word u is accepted or recognized by a finite automaton $\mathcal{A}$ if there exists a path in $\mathcal{A}$ from an initial state to a final state labelled by u. The language of words accepted by $\mathcal{A}$ is denotes by $L(\mathcal{A})$.

An infinite word w is accepted or recognized by a finite automaton $\mathcal{A}$ if there exists an infinite path in $\mathcal{A}$ starting from an initial states of $\mathcal{A}$ and using infinitely many final states of $\mathcal{A}$. In this context, a finite automaton is commonly called a Büchi automaton. The ω-language of ω-word accepted by $\mathcal{A}$ is denotes by $L_\omega(\mathcal{A})$.

If u is a finite or infinite word, $\alpha(u)$ denotes the set of letters occurring in u. This notion is extended to languages or ω-languages: $\alpha(L) = \cup_{u \in L} \alpha(u)$.

If R is a semi-commutation relation an u a finite word, we denote by $R(u) = \{xbay \mid x \in A^*, y \in A^*, (a, b) \in R \text{ and } xaby = u\}$. We denote by by $R^*(u)$ the language $\{u\} \cup \cup_{k \geq 1} R^k(u)$. By extension, for a language L, we set

$$R(L) = \cup_{u \in L} R(u) \quad \text{and} \quad R^*(L) = \cup_{u \in L} R^*(u).$$

A language (resp. ω-language) L is R-closed if $R^*(L) = L$.

2 R-shuffle Product and Finite Automata

We first extend a result of [DM97] to infinite words.

Proposition 1 *Let L_1 be a language of finite words and L_2 a language of ω-words. One has:*

$$R^*(L_1 L_2) = R^*(L_1) ⧢_R R^*(L_2).$$

Proof.

$\subseteq$: By definition of $\sqcup\!\sqcup_R$ one has $L_1L_2 \subseteq L_1 \sqcup\!\sqcup_R L_2$. Therefore, since $L_1 \subseteq R^*(L_1)$ and $L_2 \subseteq R^*(L_2)$, one has

$$L_1L_2 \subseteq R^*(L_1) \sqcup\!\sqcup_R R^*(L_2). \tag{1}$$

Now let $w \in R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)$. We claim that $R(w) \subseteq R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)$. There exists u and v such that $w \in u \sqcup\!\sqcup_R v$. Moreover, by definition of $\sqcup\!\sqcup_R$, there exist u_i's and v_i's such that $u_1v_1 \dots u_nv_n$ with $u = u_1 \dots u_n$, $v = v_1 \dots v_n$ and such that $\alpha(u_i) \times \alpha(v_j) \subseteq R$ for all $j < i$. Let $w' \in R(w)$. According the position of the rewriting process, following cases arise:

- The semi-commutation occurs *in* u_k: one has $w' = u_1v_1 \dots u_{k-1}v_{k-1}u'_kv_ku_{k+1} \dots u_nv_n$ with $u'_k \in R(u_k)$. Since $\alpha(u'_k) = \alpha(u_k)$, $w' \in R(u) \sqcup\!\sqcup_R v$. But $u \in R^*(L_1)$, therefore $w' \in R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)$.
- The semi-commutation occurs *in* v_k: similarly, one has $w' \in R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)$.
- The semi-commutation occurs *at the end of a* u_k: one has $w' = u_1v_1 \dots u_{k-1}v_{k-1}u'_kv'_ku_{k+1} \dots u_nv_n$ with $u_k = xa$, $v_k = by$, $u'_k = xb$, $v'_k = ay$ and $(a, b) \in R$. In this context, set $x_i = u_i$ and $y_i = v_i$ for $i < k$ and $x_k = u'_k$ and $y_k = v'_k$. Let also $x_i = u_{i+1}$ and $y_i = v_{i+1}$ for $i > k$. Finally, let $x_{k+1} = b$ and $y_{k+1} = b$. One has $w' = x_1y_1 \dots x_{n+1}v_{n+1}$. Moreover, $u = x_1 \dots x_{n+1}$ and $v = y_1 \dots y_n$. Now, one can easily check that $\alpha(x_i) \times \alpha(x_j) \subseteq R$ for all $j < i$. Therefore, $w' \in u \sqcup\!\sqcup_R v$. Thus $w' \in R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)$.
- The semi-commutation occurs *at the end of a* v_k: by a similar decomposition, one has $w' \in R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)$, proving the claim.

Consequently $R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)$ is R-closed. Therefore, using (1), one has

$$R^*(L_1L_2) \subseteq R^*(R^*(L_1) \sqcup\!\sqcup_R R^*(L_2)) = R^*(L_1) \sqcup\!\sqcup_R R^*(L_2).$$

$\supseteq$: By a straightforward induction, one has: for every $u \in A^*$ and every $v \in A^\omega$,

$$u \sqcup\!\sqcup_R v \subseteq R^*(uv).$$

Obviously this inclusion can be extended to languages. Thus

$$R^*(L_1) \sqcup\!\sqcup_R R^*(L_2) \subseteq R^*(R^*(L_1)R^*(L_2)).$$

Since $R^*(R^*(L_1)R^*(L_2)) = R^*(L_1L_2)$, one has

$$R^*(L_1L_2) \supseteq R^*(L_1) \sqcup\!\sqcup_R R^*(L_2),$$

which concludes the proof.

□

The above result may be easily extended to a finite product of languages by an obvious induction on the length of the product. Thanks to this result, we reduce the computation of $R^*(L_1L_2)$ to the computation of $R^*(L_1)$, $R^*(L_2)$ and of $\sqcup\!\sqcup -R$ operator. We are now interested in a procedure for computing $L_1 \sqcup\!\sqcup -RL_2$ when L_1 and L_2 are given by finite automata.

Proposition 2 *Let $\mathcal{A}_1 = (Q_1, A, E_1, I_1, F_1)$ and $\mathcal{A}_2 = (Q_2, A, E_2, I_2, F_2)$ be two finite automata on A. We define the automaton $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ by:*

- *the set of states of $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ is*

$$Q_1 \times Q_2 \times 2^A \cup Q_2$$

- *the set of initial states of $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ is*

$$I_1 \times I_2 \times \{\emptyset\},$$

- *the set of final states of $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ is F_2,*
- *the set of transitions of $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ is*

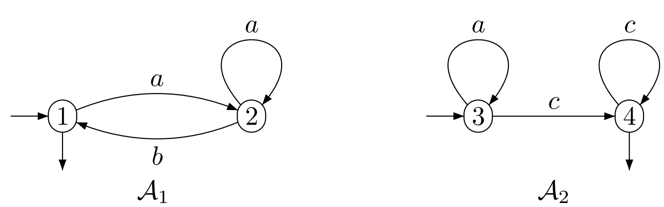
$$\begin{aligned} &\{(p,q,X) \to_a (r,q,X) \mid r \in p \cdot_{\mathcal{A}_1} a \text{ and } (a,X) \subseteq R\} \\ \cup&\{(p,q,X) \to_a (p,r,X \cup \{a\}) \mid r \in q \cdot_{\mathcal{A}_2} a\} \\ \cup&\{(p,q,X) \to_a r \mid r \in q \cdot_{\mathcal{A}_2} a, p \in F_1\} \\ \cup& E_2 \end{aligned}$$

where p, q, X, a respectively describe $Q_1, Q_2, 2^A$ and A.

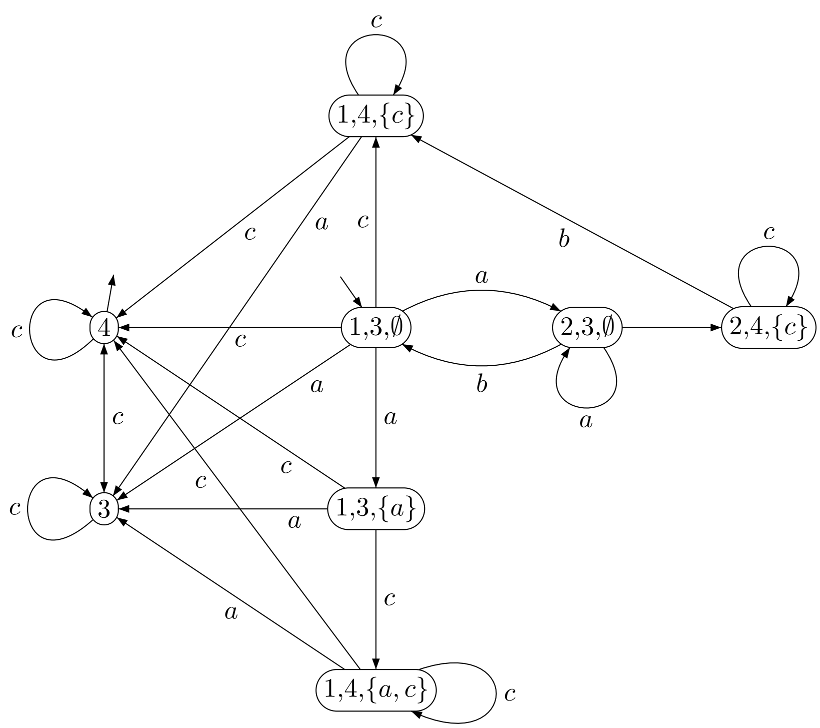
One has

$$L_\omega(\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2) = L(\mathcal{A}_1) \sqcup\!\sqcup_R L_\omega(\mathcal{A}_2).$$

Consider for example the two following automata:



The construction of $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ provides the following finite automaton:



Notice that in the construction, if $(p, q, B) \rightarrow_a (r, s, D)$ is a transition, then $B \subseteq D$.
Proof.

The proof will naturally be divided into two steps: we will first prove that $L_\omega(\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2) \subseteq L(\mathcal{A}_1) \sqcup\!\sqcup_R L_\omega(\mathcal{A}_2)$ and second that $L(\mathcal{A}_1) \sqcup\!\sqcup_R L_\omega(\mathcal{A}_2) \subseteq L_\omega(\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2)$.

To simplify notations, set

- $\mathcal{A} = \mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$,
- $G_1 = \{(p, q, X) \rightarrow_a (r, q, X) \mid r \in p \cdot_{\mathcal{A}_1} a \text{ and } (a, X) \subseteq R, p \in Q_1, q \in Q_2, X \subseteq 2^A, a \in A\}$,
- $G_2 = \{(p, q, X) \rightarrow_a (p, r, X \cup \{a\}) \mid r \in q \cdot_{\mathcal{A}_2} a, p \in Q_1, q \in Q_2, X \subseteq 2^A, a \in A\}$,
- $G_3 = \{(p, q, X) \rightarrow_a r \mid r \in q \cdot_{\mathcal{A}_2} a, p \in F_1, p \in Q_1, q \in Q_2, X \subseteq 2^A, a \in A\}$.

Note that the set of transitions of $\mathcal{A}$ is $G_1 \cup G_2 \cup G_3 \cup E_2$.

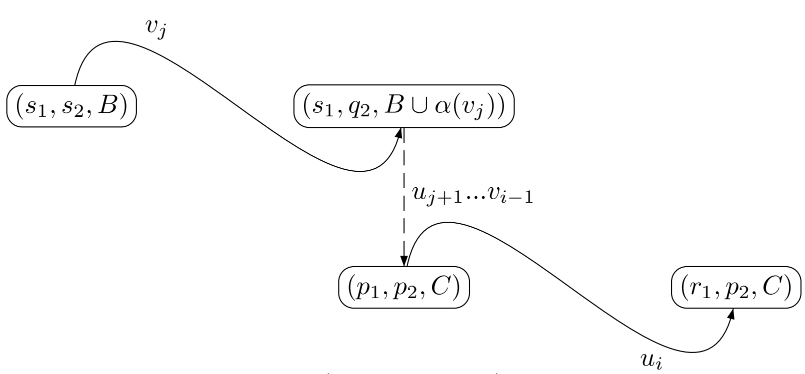
Let $w \in L_\omega(\mathcal{A})$. By definition, there exists an infinite path m in $\mathcal{A}$ labelled by w, starting from an initial state of $\mathcal{A}$ and using infinitely many final states of $\mathcal{A}$. By construction, all transitions of G_1 and G_2 are between states of $Q_1 \times Q_2 \times 2^A$, all transitions of G_3 starts form a states of $Q_1 \times Q_2 \times 2^A$ and ends in a state of Q_2, and all transitions of E_2 are between states of Q_2. Thus, since initial states of $\mathcal{A}$ are in $Q_1 \times Q_2 \times 2^A$ and final states of $\mathcal{A}$ are in Q_2, the path m can be decomposed into:

$$m = m_{\text{finite}}, t, m_\omega$$

where the path m_{finite} is a finite path using only transitions of G_1 and G_2, $t \in G_3$ and m_ω is an infinite path using only transitions of E_2.
In turn, the finite path m_{finite} can be decomposed into:

$$m_{\text{finite}} = m_1, m_2, m_3, \ldots, m_k$$

such that each m_{2i+1} $(0 \leq i \leq (k-1)/2)$ only uses transitions of G_1 and each m_{2i} $(1 \leq i \leq k/2)$ only uses transitions of G_2 (some of them may be empty). Now, let us denote by u_{i+1} the label of m_{2i+1} and v_i the label of m_{2i}. By construction, the label of m_{finite} is $u_1 v_1 u_2 \ldots u_r v_r$ ($r = k/2$ if k is even and $r = (k-1)/2$ if k is odd). We claim that for all $1 \leq j < i \leq r$, $\alpha(u_i) \times \alpha(v_j) \subseteq R$. Indeed, let $1 \leq j < i \leq r$. Assume that u_i or v_j is empty. Then $\alpha(u_i) \times \alpha(v_j) = \emptyset \subseteq R$. Assume now that u_i and v_j are both non-empty. Let (s_1, s_2, B) be the first state of m_{2j}. Since m_{2j} only uses transitions of G_2 the last state of m_{2j} is of the form $(s_1, q_2, B \cup \alpha(v_j))$. Let (p_1, p_2, C) the first state of m_{2i+1}. Since m_{2i+1} only uses transitions of G_1, its last state is of the form (r_1, p_2, C).



By construction $C = B \cup \alpha(v_j v_{j+1} \ldots v_{i-1})$. Moreover, since the path m_{2i+1} only uses transitions of G_1, each letter $a \in \alpha(u_i)$ has to satisfy $\{a\} \times C \subseteq R$. It follows that

$$\alpha(u_i) \times \alpha(v_j) \subseteq R, \tag{2}$$

proving the claim.
Now since $m = m_{\text{finite}} t m_\omega$ and since $t \in G_3$, the last state of m_{finite} is of the form (p, q, D) with $p \in F_1$. Consequently,

$$u_1 u_2 \ldots u_r \in L(\mathcal{A}_1). \tag{3}$$

Now let v be the label of t, m_ω. By construction, the path $m_2, m_4, \ldots, m_{2r}, t, m_\omega$ is labelled by $v_0 v_1 \ldots v_r v$ and is a word of $L_\omega(\mathcal{A}_2)$. Consequently, and by (2) and (3), $w \in L(\mathcal{A}_1) \sqcup\!\sqcup_R L_\omega(\mathcal{A}_2)$, proving the first step of the proof.

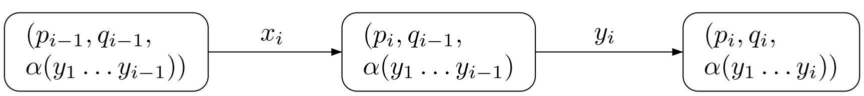
Now we can we prove that $L(\mathcal{A}_1) \sqcup\!\sqcup_R L_\omega(\mathcal{A}_2) \subseteq L(\mathcal{A})$. Let z be in $L(\mathcal{A}_1) \sqcup\!\sqcup_R L_\omega(\mathcal{A}_2)$. By definition there exist $x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n \in A^*$, $y_n \in A^\omega$, such that $x_1 x_2 \ldots x_n \in L(\mathcal{A}_1)$, $y_1 y_2 \ldots y_n \in L_\omega(\mathcal{A}_2)$ and for all $1 \leq i \leq n$ and for all $1 \leq j < i \leq n$, $\alpha(x_i) \times \alpha(y_j) \subseteq R$. Since $x_1 x_2 \ldots x_n \in L(\mathcal{A}_1)$, there exist $p_0, p_1, \ldots, p_n \in Q_1$ such that

- $p_0 \in I_1$,
- $p_n \in F_1$,
- for all $i \in \{1, \ldots, n\}$, there exists a path in $\mathcal{A}_1$ from p_{i-1} to p_i labelled by x_i.

Since $y_1 y_2 \ldots y_n \in L_\omega(\mathcal{A}_2)$, there exist $q_0, q_1, \ldots, q_n \in Q_2$ such that

- $q_0 \in I_2$,
- $q_n \in F_2$,
- for all $i \in \{1, \ldots, n-1\}$, there exists a path in $\mathcal{A}_2$ from q_{i-1} to q_i labelled by y_i.
- there exists an infinite path in $\mathcal{A}_2$ from q_{n-1} visiting infinitely many often q_n.

For all $i \in \{1, \ldots, n-1\}$, let us denote by t_i the word $y_1 \ldots y_i$. Moreover, let $t_0 = \varepsilon$. We claim that for all $i \in \{1, \ldots n\}$, there exist a path in $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ labelled by x_i from $(p_{i-1}, q_{i-1}, \alpha(t_{i-1}))$ to $(p_i, q_{i-1}, \alpha(t_{i-1}))$ and for all $i \in \{1, \ldots n-1\}$,a path in $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ labelled by y_i from $(p_i, q_{i-1}, \alpha(t_{i-1}))$ to $(p_i, q_i, \alpha(t_i))$.



Let i be in $\{1, \ldots, n\}$. Since for all j such that $1 \leq j < i$, $\alpha(x_i) \times \alpha(y_j) \subseteq R$, one has $\alpha(x_i) \times \alpha(t_{i-1}) \subseteq R$. Thus, by definition of p_{i-1}, p_i, q_{i-1} and by construction of $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$, there exists a path in $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ labelled by x_i from $(p_{i-1}, q_{i-1}, \alpha(t_{i-1}))$ to $(p_i, q_{i-1}, \alpha(t_{i-1}))$. Furthermore, by definition of q_{i-1}, p_i, q_i and by construction of $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$, there exists a path in $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ labelled by y_i from $(p_i, q_{i-1}, \alpha(t_{i-1}))$ to $(p_i, q_i, \alpha(t_i))$, proving the claim.

Now, let a be the first letter of y_n and set $y_n = a y_n'$. By definition of q_{n-1} and q_n there exists a state $q_n' \in Q_2$ such that $(q_{n-1}, a, q_n) \in E_2$ and such that there exists an infinite path from q_n' labelled by y_n' and visiting infinitely many often q_n'. Since p_n is final in $\mathcal{A}_1$, there exists in $\mathcal{A}$ a transition from $(p_n, q_{n-1}, \alpha(y_1 \ldots y_{n-1})$ to q_n' labelled by a. Consequently, there exists an infinite path in $\mathcal{A}$ from $(p_n, q_{n-1}, \alpha(y_1 \dots y_{n-1}))$ labelled by y_n and visiting infinitely many often q_n. It results that $z \in L(\mathcal{A})$, which concludes the proof. □

3 Permutation Rewriting and Polynomial Closure of Commutative Regular Languages

In this section the main stability results of the paper are proved.

A regular language on finite words is commutative if and only if its minimal automaton is a diamond automaton. It is obvious that all languages and ω-languages accepted by diamond automata are R-closed for all semi-commutation relations.

The class $\mathrm{Pol}\mathcal{C}$ (polynomial closure of commutative regular languages) is composed of finite union of languages of the form $L_0a_0L_1a_1 \dots a_kL_k$ where the a_i's are letters and the L_i's are commutative regular languages.

One has the following result [CHM03].

Theorem 3 *The class* $\mathrm{Pol}\mathcal{C}$ *is closed under semi-commutation.*

Recall that $\omega - \mathrm{Pol}C$ is the class of ω-languages which are a finite union of languages of the form

$$L_0^* a_1 L_2 \dots a_{k-1} L_{k-1} a_k L_k$$

where the a_i's are letters of A and the L_i's $(i < k)$ are commutative regular languages and L_k is accepted by a diamond Büchi automaton.

Following results in [BMT07], we also introduce the following class of regular ω-languages which is the infinite words version of APC or of languages of level 3/2 in Straubing's hierarchy [Thé81, Str85].

Proposition 4 *Let* L *be a regular* ω*-language.The following propositions are equivalent:*

(1) L *is a finite union of languages of the form*

$$A_0^* a_1 A_2^* \dots a_{k-1} A_{k-1}^* a_k A_k^\omega$$

where the a_i*'s are letters of* A *and the* A_i*'s are subsets of* A*.*

(2) L *is recognized by a partially ordered Büchi automaton.*

This class of ω*-languages is called* ω*-alphabetic pattern constraints and is denoted* ω*-APC.*

The proof is obvious and left to the reader.

Theorem 5 *The classes* ω*-APC and* $\omega - \mathrm{Pol}\mathcal{C}$ *are closed under semi-commutations.*

The proof of Theorem 5 is obtained thanks to the sequence of lemmas below.

Lemma 6 *Let $\mathcal{A} = (Q, A, E, I, F)$ be a finite automaton, L_1, L_2 be two languages on A and R a semi-commutation relation over A. The following equality holds:*

$$L_1L_2 ⧢_R L(\mathcal{A}) = \cup_{q\in Q}((L_1 ⧢_R (L(\mathcal{A}_{I,q}) \cap B^*))((L_2 \cap C^*) ⧢_R L_\omega(\mathcal{A}_{q,F}))$$

where the union is taken for all subsets B and C of A such that $C \times B \subseteq R$.

Proof. Let $q \in Q$ and $u \in ((L_1 ⧢_R (L(\mathcal{A}_{I,q}) \cap B^*))((L_2 \cap C^*) ⧢_R L_\omega(\mathcal{A}_{q,F}))$, with $C \times B \subseteq R$. Then u can be decomposed into:

$$u = x_1y_1 \dots x_ny_nz_1t_1 \dots z_kt_k$$

such that

(1) $x_1 \dots x_n \in L_1$, $y_1 \dots y_n \in L(\mathcal{A}_{I,q}) \cap B^*$,

(2) for all $1 \leq j < i \leq n$, $\alpha(x_i) \times \alpha(y_j) \subseteq R$,

(3) $z_1 \dots z_k \in L_2 \cap C^*$, $t_1 \dots t_k \in L(\mathcal{A}_{q,F})$,

(4) for all $1 \leq j < i \leq k$, $\alpha(z_i) \times \alpha(t_j) \subseteq R$,

Since $C\times B \subseteq R$ and by (1) and (3), for all $1 \leq i \leq n$ and for all $1 \leq j \leq k$, $\alpha(z_j)\times\alpha(y_j) \subseteq R$. Consequently and by (2) and (4), $u \in L_1L_2 ⧢_R L_\omega(\mathcal{A})$.
Conversely, let $u \in L_1L_2 ⧢_R L_\omega(\mathcal{A})$. By definition of the R-shuffle, there exist $x_1, \dots, x_{n-1}, y_1 \dots, y_n \in A^*$ and $x_n \in A^\omega$ such that

(5) $u = y_1x_1 \dots y_nx_n$

(6) for all $1 \leq j < i \leq n$, $\alpha(x_j) \times \alpha(y_i) \subseteq R$,

(7) $x_1 \dots x_n \in L_\omega(\mathcal{A})$,

(8) $y_1 \dots y_n \in L_1L_2$.

Statement (8) implies that there is $1 \leq k \leq n$ such that y_k may be decomposed into $y_k = st$, with $s, t \in A^*$ and $y_1 \dots y_{k-1}s \in L_1$ and $ty_{k+1} \dots y_n \in L_2$. Statement (7) implies that there exists a state q such that $x_1 \dots x_k \in L(\mathcal{A}_{I,q})$ and $x_{k+1} \dots x_n \in L_\omega(\mathcal{A}_{q,F})$. Now, by (5) and (6),

$$y_1x_1y_2 \dots x_{k-1}y_{k-1}s \in L_1 ⧢_R (L(\mathcal{A}_{I,q}) \cap \alpha(x_1 \dots x_{k-1})^*)$$

and

$$tx_{k+1}y_{k+1} \dots y_nx_n \in (L_2 \cap \alpha(ty_{k+1} \dots y_n)^*) ⧢_R L(\mathcal{A}_{q,F})$$

By (6), $\alpha(x_1 \dots x_k) \times \alpha(ty_{k+1} \dots y_n) \subseteq R$, which concludes the proof. □

Lemma 7 *Let $\mathcal{A}_1 = (Q_1, A, E_1, I_1, F_1)$ and $\mathcal{A}_2 = (Q_2, A, E_2, I_2, F_2)$ be two finite automata and R a semi-commutation relation over A. If $\mathcal{A}_1$ and $\mathcal{A}_2$ are diamond automata, then $L_\omega(\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2) \in \omega - \mathrm{Pol}\mathcal{C}$.*

Proof. Let $\mathcal{A} = (Q, A, E, I, F)$ be the trim automaton obtained from $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$. For all subsets B of $\alpha(L(\mathcal{A}_2))$, we denote by Q_B the subset $\{(q_1, q_2, B) \mid q_1 \in Q_1, q_2 \in Q_2\}$ of Q and by E_B the subset $E \cap Q_B \times A \times Q_B$ of E.

Let $t = ((p, q, C), a, (p', q', D)) \in E \setminus \cup_{B \subseteq A} E_B$. We claim that there is no loop in $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ using t: since $C \subsetneq D$ all states accessible from (p', q', D) are of the form (r, s, B), with $D \subseteq B$.

Each successful path m in $\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2$ can be decomposed into:

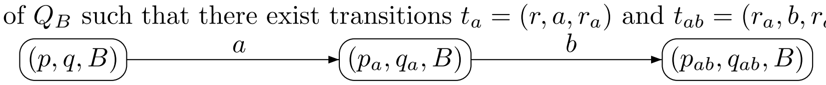
$$m = m_0, t_1, m_1, t_2, \dots, t_n, m_n, t, m_\omega$$

with $t_i \in E \setminus \cup_{B \subseteq A} E_{\overline{B}}$ and m_i only using transitions of E_{B_i}, for all $0 \leq i \leq n$. Using the above claim, we have $n \leq |E \setminus \cup_{B \subseteq A} E_B|$. Consequently, $L(\mathcal{A}_1 \sqcup\!\sqcup_R \mathcal{A}_2)$ is a finite union of languages of the form:

$$L_0 a_1 L_1 a_2 \dots a_n L_n a L$$

where the a_i's are letters and the L_i's are accepted by finite automata whose graphs of transitions are (Q_{B_i}, E_{B_i}), a is the label of t and L is accepted by finite automata whose graphs of transitions (Q_2, E_2).

By Lemma 1, it remains to prove that the L_i's are commutative languages and that L is accepted by a diamond automaton. Since $\mathcal{A}_2$ is a diamond automaton, L is accepted by a diamond automaton. Now, let $B \subseteq A$, we prove that the monoid of transitions generated by (Q_B, E_B) is commutative. Let $r = (p, q, B)$, $r_a = (p_a, q_a, B)$ and $r_{ab} = (p_{ab}, q_{ab}, B)$ be three states of Q_B such that there exist transitions $t_a = (r, a, r_a)$ and $t_{ab} = (r_a, b, r_{ab})$ in E_B.



With the notation of the proof of Proposition 2, the following cases occur:

- $t_a, t_{ab} \in G_1$. Since $\mathcal{A}_1$ is minimal and since $L(\mathcal{A}_1)$ is commutative, the transition monoid of $\mathcal{A}_1$ is commutative. Thus there exists p_b in Q_1 such that $p \cdot b = p_b$ and $p_b \cdot a = p_{ab}$. Moreover, since t_a and t_b belong to G_1, $\{a\} \times B \subseteq R$ and $\{b\} \times B \subseteq R$. Consequently, $(r, b, (p_b, q, B))$ and $((p_b, q, B), a, r_{ab})$ are in $G_1 \cap E_B$. It follows that $r_{ab} \in r \cdot ba$.
- $t_a, t_{ab} \in G_2$. By a similar argument on $\mathcal{A}_2$, one has $r_{ab} \in r \cdot ba$.
- $t_a \in G_1, t_{ab} \in G_2$. Thus $q_a = q$ and $p_{ab} = p_a$. Consequently $(r, b, (p, q_{ab}, B)) \in G_2 \cap E_B$ and $((p, q_{ab}, B), a, r_{ab}) \in G_1 \cap E_B$. It follows that $r_{ab} \in r \cdot ba$.
- $t_a \in G_2, t_{ab} \in G_1$. By a similar argument on $\mathcal{A}_2$, one has $r_{ab} \in r \cdot ba$.

Consequently $r \cdot ab \subseteq r \cdot ba$. Since the roles of a and b are symmetric, $r \cdot ba \subseteq r \cdot ab$. Therefore, the monoid of transitions generated by (Q_B, E_B) is commutative, which concludes the proof. □

Lemma 8 *Let K be a language of* Pol$\mathcal{C}$ *and $\mathcal{A}$ a diamond automaton. Then $K \sqcup\!\sqcup_R L(\mathcal{A})$ belongs to $\omega -$* Pol$\mathcal{C}$.

Proof. For each regular language K and each ω-language L, one has: if $\varepsilon \in K$, then $KL = L \cup_{a\in A} (Ka^{-1})aL$, and if $\varepsilon \notin L$, then $KL = \cup_{a\in A}(Ka^{-1})aL$, with $Ka^{-1} = \{v \in A^* \mid va \in K\}$. Moreover, since the class of languages accepted by diamond automata forms a variety of regular languages, if Ka^{-1} is accepted by a diamond automaton too.

Now Lemma can be proved by a direct trivial induction using Lemma 7. □

The same proof works for the following lemma.

Lemma 9 *Let K be an APC language and L a language of the form B^ω, where $B \subseteq A$. Then $K \sqcup\!\sqcup_R L$ belongs to ω-APC.*

One can now prove Theorem 5.

Proof. Let R be a semi-commutation relation.

We just give the proof for $\omega -$ Pol$\mathcal{C}$ languages. The same proof works for $\omega-$APC.

Since for all sets H and I of A^ω, $R^*(H \cup I) = R^*(H) \cup R^*(I)$, we only have to prove the result for languages of the form $L = L_0a_1L_1 \ldots a_kL_ka_{k+1}L_{k+1}$, where the L_i's are accepted by diamond automata and the a_i's are letters.

Now, by Proposition 1, one has

$$R^*(L) = R^*(L_0a_1L_1 \ldots a_kL_ka_{k+1}) \sqcup\!\sqcup_R L_{k+1}.$$

Using Theorem 3, one has $R^*(L_0a_1L_1 \ldots a_kL_ka_{k+1})$ belongs to Pol$\mathcal{C}$, and we conclude by Lemma 8. □

4 Conclusion

The results presented in this paper improve recent works by Bouajjani, Muscholl and Touili on one hand, and by Cécé, Héam and Mainier on the other hand, by extending them to infinite words.

We intend to investigate practical applications of this work, particularly for HMSC's formal verification. As far as we know, several connected theoretical problems remains open: are the classes $\omega-$APC, $\omega -$ Pol$\mathcal{C}$ and Pol$\mathcal{C}$ decidable (the class APC is decidable [Arf91]). Another difficult related problem is, given a language of $\omega -$ Pol$\mathcal{C}$, to decompose it into a finite union of products of languages accepted by diamond automata. The same problem faces for Pol$\mathcal{C}$, while an inefficient algorithm exists for APC. This kind of problems generally requires deep semi-groups theory arguments, see [Pin87, Pin94] for instance.

References

[AAB99] P. Abdulla, A. Annichini, and A. Bouajjani. Algorithmic verification of lossy channel systems: An appliction to the bounded retransmission protocol. In *TACAS'99*, volume 1579 of *Lecture Notes in Computer Science*, pages 208–222, 1999.

[ABJ98] P. A. Abdulla, A. Bouajjani, and B. Jonsson. On-the-fly analysis of systems with unbounded, lossy FIFO channels. In *CAV'98*, volume 1427 of *Lecture Notes in Computer Science*, pages 305–322, 1998.

[AJNd03] Parosh Aziz Abdulla, Bengt Jonsson, Marcus Nilsson, and Julien d'Orso. Algorithmic improvements in regular model checking. In *CAV'03*, LNCS, 2003.

[Arf91] M. Arfi. Opération polynomiales et hiérarchies de concaténation. *Theoretical Computer Science*, 91:71–84, 1991.

[BB99] J. Berstel and L. Boasson. Shuffle factorization is unique. Technical report, LIAFA - Université Paris 7, 1999.

[BBC+06] Jean Berstel, Luc Boasson, Olivier Carton, Bruno Petazzoni, and Jean-Eric Pin. Operations preserving regular languages. *Theor. Comput. Sci*, 354(3):405–420, 2006.

[Ber79] Jean Berstel. *Transductions and Context-Free Languages*. B.G. Teubner, Stuttgart, 1979.

[BG96] B. Boigelot and P. Godefroid. Symbolic verification of communication protocols with infinite state spaces using QDDs. In *Proc. of* 8^{th} *CAV (August), USA*, volume 1102, pages 1–12. LNCS, 1996.

[BMT01] A. Bouajjani, A. Muscholl, and T. Touili. Permutation rewriting and algorithmic verification. In *LICS'01*, IEEE Computer Society, pages 399–408, 2001.

[BMT07] Ahmed Bouajjani, Anca Muscholl, and Tayssir Touili. Permutation rewriting and algorithmic verification. *Information and Computation*, 205(2):199–224, 2007.

[Brz76] J.A. Brzozowski. Hierarchies of aperiodic languages. *Informatique théorique et Application/Theoretical Informatics and Applications*, 10:33–49, 1976.

[BS73] J.A. Brzozowski and I. Simon. Characterizations of locally testable languages. *Discrete Mathematics*, 4:243–271, 1973.

[BW98] Bernard Boigelot and Pierre Wolper. Verifying systems with infinite but regular state spaces. In *CAV'98*, volume 1427 of *LNCS*, pages 88–97, June 1998.

[CHM03] Gérard Cécé, Pierre-Cyrille Héam, and Yann Mainier. Efficiency of automata in semi-commutation verification techniques, 2003. To appear in ITA-RAIRO.

[DM97] V. Diekert and V. Métivier. *Partial Commutation on Traces*, volume III of *Handbook on Formal Languages*. Springer, 1997.

[ES98] Z. Esik and I. Simon. Modeling literal morphisms by shuffle. *Semigroup Forum*, 56:225–227, 1998.

[GM05] Blaise Genest and Anca Muscholl. Message sequence charts: A survey. In *ACSD*, pages 2–4. IEEE Computer Society, 2005.

[GP03] A. Cano Gomez and J.-E. Pin. On a conjecture of schnoebelen,. In *DLT'03*, volume 2710 of *Lecture Notes in Computer Science*, pages 35–54, 2003.

[GP04] A. Cano Gomez and J.-E. Pin. Shuffle on positive varieties of languages. *Theoretical Computer Science*, 2004. to appear.

[HU80] J. Hopcroft and J. Ullman. *Introduction to automata theory, languages, and computation*. Addison-Wesley, 1980.

[LS98] D. Lugiez and Ph. Schnoebelen. The regular viewpoint on pa-processes. In *9th Int. Conf. Concurrency Theory (CONCUR'98)*, volume 1466 of *Lecture Notes in Computer Science*. Springer, 1998.

[MP01] Anca Muscholl and Doron Peled. From finite state communication protocols to high-level message sequence charts. In Fernando Orejas, Paul G. Spirakis, and Jan van Leeuwen, editors, *ICALP*, volume 2076 of *Lecture Notes in Computer Science*, pages 720–731. Springer, 2001.

[NRR+94] M. Nivat, G.D.S. Ramkumar, C. Pandu Rangan, A. Saoudi, and R. Sundaram. Efficient parallel shuffle recognition. *Parallel Processing Letters.*, 4:455–463, 1994.

[Per78] J.-F. Perrot. Variété de langages et opérations. *Theoretical Computer Science*, 7:197–210, 1978.

[Pin87] Jean-Eric Pin. On the language accepted by finite reversible automata. In Thomas Ottmann, editor, *Automata, Languages and Programming, 14th International Colloquium*, volume 267 of *Lecture Notes in Computer Science*, pages 237–249, Karlsruhe, Germany, 13–17 July 1987. Springer-Verlag.

[Pin94] J.-E. Pin. Polynomial closure of group languages and open sets of the Hall topology. *Lecture Notes in Computer Science*, 820:424–432, 1994.

[PMR98] B. Pradeep, C. Murthy, and S. Ram. A constant time string shuffle algorithm on reconfigurable meshes. *Int. J. Comput. Math.*, 68:251–259, 1998.

[PP04] D. Perrin and J.-E. Pin. *Infinite Words*, volume 141. Elsevier, 2004.

[PW97] J.-E. Pin and P. Weil. Polynomial closure and unambiguous product. *Theory Comput. Systems*, 30:1–39, 1997.

[Rad79] D.E. Radford. A natural ring basis for shuffle algebra and an application to group schemes. *Journal of Algebra*, 58:432–454, 1979.

[Sch99] Ph. Schnoebelen. Decomposable regular languages and the shuffle operator. *EATCS Bull.*, 67:283–289, 1999.

[Spe86] J.-C. Spehner. Le calcul rapide des mélanges de deux mots. *Theoretical Computer Science*, 47:181–203, 1986.

[Str85] H. Straubing. Finite semigroups varieties of the form $\mathbf{V}*\mathbf{D}$. *Journal of Pure and Applied Algebra*, 36:53–94, 1985.

[Tho82] W. Thomas. Classifying regular events in symbolic logic. *Journal of Computer and System Science*, 25:360–375, 1982.

[Thé81] D. Thérien. Classification of finite monoids: the language approach. *Theoretical Computer Science*, 14:195–208, 1981.

[Tou01] T. Touili. Regular model checking using widening techniques. In *1st Vepas Workshop*, volume 50 of *Electronic Notes in TCS*, 2001.

[TT02] P. Tesson and D. Thérien. Diamonds are forever: the variety da. In *International Conference on Semigroups, Algorithms, Automata and Languages*, 2002.

Unité de recherche INRIA Lorraine
LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399