The expressive power of existential first order sentences of Büchi's sequential calculus

Jean-Eric Pin
LITP/IBP, CNRS, Université Paris VI,
4 Place Jussieu, 75252 Paris Cedex 05,
FRANCE

March 12, 2009

Abstract

The aim of this paper is to study the first order theory of the successor, interpreted on finite words. More specifically, we complete the study of the hierarchy based on quantifier alternations (or Σ_n-hierarchy). It was known (Thomas, 1982) that this hierarchy collapses at level 2, but the expressive power of the lower levels was not characterized effectively. We give a semigroup theoretic description of the expressive power of Σ_1, the existential formulas, and $\mathcal{B}\Sigma_1$, the boolean combinations of existential formulas. Our characterization is algebraic and makes use of the syntactic semigroup, but contrary to a number of results in this field, is not in the scope of Eilenberg's variety theorem, since $\mathcal{B}\Sigma_1$-definable languages are not closed under residuals.

An important consequence is the following: given one of the levels of the hierarchy, there is polynomial time algorithm to decide whether the language accepted by a deterministic n-state automaton is expressible by a sentence of this level.

1 The sequential calculus

The connections between formal languages and mathematical logic were first studied by Büchi [5]. But although Büchi was primarly interested in infinite words, we will consider only finite words in this paper.

Büchi's sequential calculus is a logical formalism to specify some properties of a finite word, for instance "the factor *bba* occurs three times in the word, but the factor *bbb* does not occur". Thus, each logical sentence of this calculus defines a language, namely the set of all words that satisfy the property expressed by the formula. For instance, in our example, this language would be $A^*bbaA^*bbaA^*bbaA^* \setminus A^*bbbA^*$, where $A = \{a, b\}$ denotes the alphabet.

More formally, to each word $u \in A^+$ is associated a structure

$$\mathcal{M}_u = (\{1, 2, \ldots, |u|\}, S, (R_a)_{a \in A})$$

where S denotes the successor relation on $\{1, 2, \ldots, |u|\}$ and R_a is set of all i such that the i-th letter of u is an a. For instance, if $A = \{a, b\}$ and $u = abaab$, then $R_a = \{1, 3, 4\}$ and $R_b = \{2, 5\}$. The logical language appropriate to such models has S and the R_a's as non logical symbols, and formulas are built in the standard way by using these non-logical symbols, variables, boolean connectives, equality between elements (positions) and quantifiers. Note that the symbol $<$ is not used in this logic. We shall use the notations F_1 (resp. F_2) for the set of first order (resp. second order) formulas with signature $\{S, (R_a)_{a \in A}\}$.

Given a sentence φ, we denote by $L(\varphi)$ the set of all words which satisfy φ, when words are considered as models. It is a well known result of Büchi that monadic second order sentences exactly define the recognizable (or regular) languages. That is, for each monadic second order sentence φ, $L(\varphi)$ is a recognizable language and, for every recognizable language L, there exists a monadic second order sentence φ such that $L(\varphi) = L$. Actually, monadic second order logic constitutes a border line in the study of the sequential calculus. Beyond that border, one enters the hard world of complexity classes [7].

2 First order

The expressive power of F_1, the set of first order formulas with signature $\{S, (R_a)_{a \in A}\}$ was first studied by Thomas [24].

2.1 The combinatorial description

Some definitions from language theory are in order to state the result of Thomas. First, we will make a distinction between *positive boolean operations* on languages, that comprise finite union and finite intersection and *boolean operations* that comprise finite union, finite intersection and complement. Given a word x and a positive integer k, it is not very difficult to express in F_1 a property like "a factor x occurs at least k times". Let us denote by $F(x, k)$ the language defined by this property. A language L of

A^+ is *strongly threshold locally testable* (STLT for short) if it is a boolean combination of sets of the form $F(x, k)$ where $x \in A^+$ and $k > 0$. It is *threshold locally testable* (TLT) if it is a boolean combination of sets of the form uA^*, A^*v or $F(x, k)$ where $u, v, x \in A^+$ and $k > 0$. Note that uA^* (resp. A^*v) is the set of words having u as a prefix (resp. v as a suffix), a property that can also be expressed in F_1. The classes of *positively strongly locally threshold testable* (PSTLT) and *positively threshold locally testable* (PTLT) languages are defined similarly, by replacing "boolean combination" by "positive boolean combination" in the definition[1]. Thomas proved the following theorem.

Theorem 2.1 *A language is F_1-definable if and only if it is TLT.*

In fact, this result is a particular instance of the general fact that first order formulas can express only local properties [9, 25, 26].

Theorem 2.1 gave a combinatorial description of the F_1-definable languages but also led to the next question : given a finite deterministic automaton $\mathcal{A}$, is it decidable whether the language accepted by $\mathcal{A}$ is F_1-definable?

2.2 The semigroup approach

This problem was solved positively by semigroup-theoretic methods. Let L be a language of A^+. The *syntactic congruence of* L is the congruence $\sim_L$ on A^+ defined by $u \sim_L v$ if and only if, for every $x, y \in A^*$,

$$xuy \in L \iff xvy \in L$$

The quotient semigroup $S(L) = A^+/{\sim_L}$ is called the *syntactic semigroup* of L. It is also equal to the transition semigroup of the minimal automaton of $\mathcal{A}$. It follows that a language is recognizable if and only if its syntactic semigroup is finite. The quotient morphism $\eta : A^+ \to S(L)$ is called the *syntactic morphism* and the subset $P = \eta(L)$ of $S(L)$ is the *syntactic image* of L. See [14] for more details.

Recall that a finite semigroup S is *aperiodic* if there exists an integer $n \geq 0$ such that, for each $s \in S$, $s^n = s^{n+1}$. Another important property was introduced by Thérien and Weiss [23]. If e and f are idempotents[2] of S, and if r, s and t are elements of S, then $erfsetf = etfserf$.

[1] The reader is referred to [3] or to [21, p. 47] for an explanation of this terminology.

[2] An element $e \in S$ is idempotent if $e^2 = e$. One can show that a non empty finite semigroup contains at least one idempotent.

It is easier to remember this condition in terms of categories (there are also good mathematical reasons to do so). The Cauchy category of a finite semigroup S is defined as follows: the objects are the idempotents of S and, if e and f are idempotents, the arrows from e to f are the triples (e, s, f), such that $s = es = sf$. Composition of arrows is defined in the obvious way:

$$(e, s, f)(f, t, g) = (e, st, g)$$

Thus the condition above can be simply written

$$pqr = rqp \qquad (C)$$

where p and r are coterminal arrows, say, from e to f, and q is an arrow from f to e.

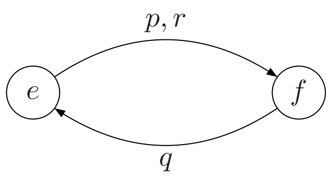


Figure 2.1: The condition $pqr = rqp$.

Thérien and Weiss did not explicitely mention the TLT languages in their paper but nevertheless gave the main argument of the proof of the following theorem.

Theorem 2.2 *A language is TLT if and only if its syntactic semigroup S is aperiodic and satisfies the condition (C).*

The link between the papers [24] and [23] was first observed in [2]. A complete proof of both results can also be found in the elegant book of Straubing on circuit complexity [21]. We complete these results by analyzing the complexity of the algorithm. More precisely, we prove the following result.

Theorem 2.3 *There is a polynomial time algorithm to decide whether the language recognized by a deterministic n-state automaton is F_1-definable.*

Proof. (Sketch) Testing for aperiodicity is PSPACE-complete [6], but it suffices to test whether the language is of "dot-depth one", which can be done in polynomial time [20]. Condition (C) can also be tested in polynomial time. It suffices to see if, for every configuration of the form represented in

Figure 2.2, in which e, f, p, q, r and y are paths in the given automaton, $q_1 \in F$ if and only if $q_2 \in F$.

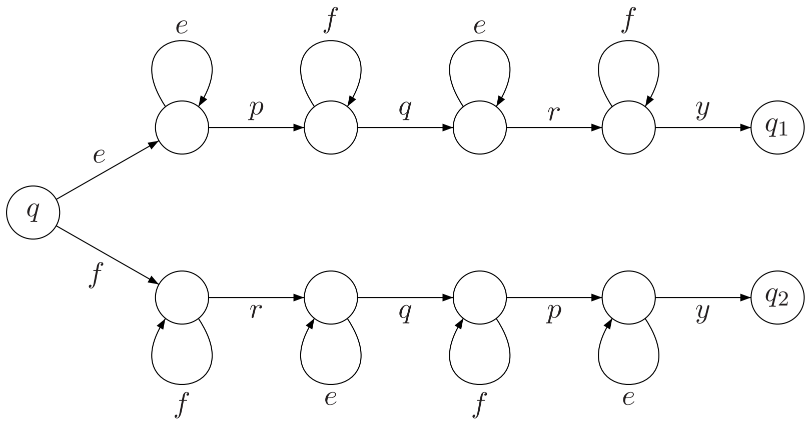


Figure 2.2: Testing Condition (C).

This can be easily tested in polynomial time. □

Theorem 2.3 is in contrast with the corresponding result for the first order logic of the binary relation $<$, interpreted as the natural order on the integers. For this logic, McNaughton and Papert [11] gave a combinatorial description (the star-free languages) and Schützenberger [18] gave an algebraic characterization (the syntactic semigroup is aperiodic), but it was shown in [6] that the corresponding algorithm is PSPACE-complete.

3 Inside first order

The details of the landscape can be refined by considering the Σ_n-hierarchy of first order logic. It was shown by Thomas [24] that any F_1-definable language can also be defined by a Σ_2-sentence, that is, a sentence of the form

$$\exists x_1 \ \cdots \ \exists x_n \ \forall y_1 \ \cdots \ \forall y_m \ \varphi(x_1, \cdots, x_n, y_1, \cdots, y_m)$$

where φ is quantifier-free. Recall that a Σ_1-formula is of the form

$$\exists x_1 \ \cdots \ \exists x_n \ \varphi(x_1, \cdots, x_n)$$

where φ is quantifier-free. Denote by Σ_1 the set of Σ_1-formulas and by $\mathcal{B}\Sigma_1$ the set of boolean combinations[3] of Σ_1-formulas. The expressive power of

[3]boolean operations on formulas comprise conjunction, disjunction and negation.

Σ_1 and $\mathcal{B}\Sigma_1$ was still to be characterized. The following result was proved in [2, 3] by using Ehrenfeucht-Fraïssé games [21].

Theorem 3.1 *A language is $\mathcal{B}\Sigma_1$-definable if and only if it is STLT.*

The proof can be easily adapted to obtain a characterization of the Σ_1-definable languages

Theorem 3.2 *A language is Σ_1-definable if and only if it is PSTLT.*

These results complete the combinatorial description of the Σ_n-hierarchy, but do not solve the decidability questions: given a finite deterministic automaton $\mathcal{A}$, is it decidable whether the language accepted by $\mathcal{A}$ is $\mathcal{B}\Sigma_1$-definable (resp. Σ_1-definable)?

The main result of this paper provides a positive answer to these questions. Let S be a finite semigroup. We denote by S^1 the monoid equal to S if S has an identity, and to $S \cup \{1\}$, where 1 is a new identity, otherwise. Two elements s and r of S are said to be $\mathcal{J}$-equivalent (notation $s \mathrel{\mathcal{J}} r$) if they generate the same ideal, that is, if there exists $x, y, u, v \in S^1$ such that $usv = r$ and $xry = s$. Let $\equiv$ be the coarsest equivalence relation on S satisfying the two following conditions

(1) for all $s, r \in S$, $s \mathrel{\mathcal{J}} r$ implies $s \equiv r$,

(2) for all idempotents e, f of S, $esfre \equiv fresf$

We say that a subset P of S *saturates* the $\equiv$-classes if, for all $s, r \in S$, $s \in P$ and $s \equiv r$ imply $r \in P$.

Theorem 3.3 *Let L be a recognizable language, S its syntactic semigroup and P its syntactic image. The following conditions are equivalent:*

(1) L is $\mathcal{B}\Sigma_1$-definable,

(2) L is STLT,

(3) S is aperiodic and satisfies (C), and P saturates the $\equiv$-classes.

For the PSTLT languages, the syntactic semigroup does not suffice, and we need the ordered syntactic semigroup, introduced in [17]. Let L be a language of A^+ and let $A^+ : \eta \to S(L)$ be its syntactic morphism. Define a relation $\preceq_L$ on A^+ by setting $u \preceq_L v$ if and only if, for every $x, y \in A^*$,

$$xvy \in L \Rightarrow xuy \in L$$

Then $\preceq_L$ is a reflexive and transitive relation such that $u \sim_L v$ if and only if $u \preceq_L v$ and $v \preceq_L u$. It follows that there is a well defined partial order on $S(L)$ defined by $\eta(u) \leq \eta(v)$ if and only if $u \preceq_L v$. This order is stable under product: if $s \leq r$ and $s' \leq r'$, then $ss' \leq rr'$. The ordered semigroup $(S(L), \leq)$ is called the *ordered syntactic semigroup* of L.

To each idempotent e is associated the subsemigroup eSe of S, defined by $eSe = \{ese \mid s \in S\}$. This is in fact a monoid, with e as an identity, called the *local submonoid* of e. Now, e is called a *local maximum* if, for every $s \in S$, $ese \leq e$. We can now formulate our characterization of the PSTLT languages.

Theorem 3.4 *Let L be a recognizable language, let S be its ordered syntactic semigroup and let P be its syntactic image. The following conditions are equivalent:*

(1) L is Σ_1-definable

(2) L is PSTLT,

(3) S satisfies (C), each idempotent of S is a local maximum and P saturates the equivalence $\equiv$.

The proof of Theorem 2.3 can be adapted to the case of $\mathcal{B}\Sigma_1$-formulas.

Corollary 3.5 *There is a polynomial time algorithm to decide whether the language recognized by a deterministic n-state automaton is $\mathcal{B}\Sigma_1-$ (resp. Σ_1-) definable.*

4 Three examples

Example 4.1 Let $A = \{a, b\}$ and let $L = a^*ba^*$. Then L is recognized by the automaton shown in Figure 4.3.

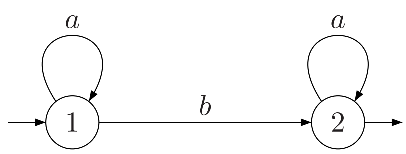


Figure 4.3: The minimal automaton of a^*ba^*.

The transitions and the relations defining the syntactic semigroup S of L are given in the following tables

	a	b	bb
1	1	2	–
2	2	–	–

$$a = 1$$
$$b^2 = 0$$

Thus $S = \{1, b, 0\}$ and $E(S) = \{1, 0\}$. The syntactic order is defined by $b \leq 0$ and $1 \leq 0$. The local semigroups are $0S0 = \{0\}$ and $1S1 = S$. The latter is not idempotent, since $b^2 \neq b$. Therefore, L is not locally testable. On the other hand, the Cauchy category of $S(L)$, represented in Figure 4.4, satisfies the condition $pqr = rqp$.

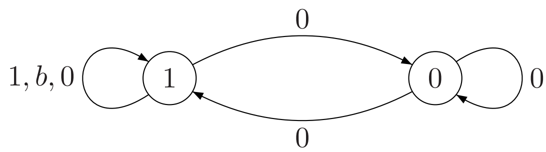


Figure 4.4: The graph of S.

Therefore L is TLT. The syntactic image of L is $P = \{b\}$, which saturates the $\equiv$-classes. Thus L is STLT. This can be seen directly in this case, since L is the set of all words containing exactly one occurrence of b. However, L is not PSTLT since, in the local semigroup $1S1 = S$, 1 is not the top element.

Example 4.2 Let $A = \{a, b, c\}$, and let $L = c(ab)^* \cup c(ab)^*a$. Then L is recognized by the following automaton.

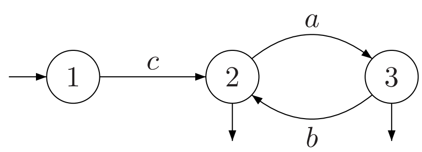


Figure 4.5: An automaton recognizing L.

The transitions and the relations defining the syntactic semigroup S of L are given in the following tables

	a	b	c	aa	ab	ba	ca
1	–	–	2	–	–	–	3
2	3	–	–	–	2	–	–
3	–	2	–	–	–	3	–

$$a^2 = b^2 = c^2 = ac = bc = cb = 0$$
$$aba = a$$
$$bab = b$$
$$cab = c$$

The $\mathcal{J}$-class structure is represented in the following diagram, where the grey box is the image of L.

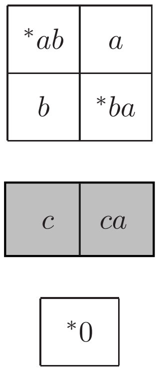


Figure 4.6: The $\mathcal{J}$-class structure.

Thus P saturates the $\mathcal{J}$-classes, and L is SLT. In fact, $L = A^*cA^* \backslash (A^*aaA^* \cup A^*acA^* \cup A^*bbA^* \cup A^*bcA^* \cup A^*cbA^* \cup A^*ccA^*)$.

Example 4.3 Let $A = \{a, b\}$, and let $L = (1 + b)a(ba)^*b^2b^*a(ba)^*(1 + b) \cup b^2b^*a(ba)^*b^2b^*$. The transitions and the relations defining the syntactic semigroup S of L are given in the following tables

Elements	1	2	3	4	5	6	7	8	9	10	11
a	6	10	0	7	10	0	0	6	7	0	6
b	11	2	3	3	0	8	4	2	9	5	9
aa	0	0	0	0	0	0	0	0	0	0	0
ab	8	5	0	4	5	0	0	8	4	0	8
ba	6	10	0	0	0	6	7	10	7	10	7
bb	9	2	3	3	0	2	3	2	9	0	9
abb	2	0	0	3	0	0	0	2	3	0	2
bab	8	5	0	0	0	8	4	5	4	5	4
bba	7	10	0	0	0	10	0	10	7	0	7
$abba$	10	0	0	0	0	0	0	10	0	0	10
$babb$	2	0	0	0	0	2	3	0	3	0	3
$bbab$	4	5	0	0	0	5	0	5	4	0	4
$abbab$	5	0	0	0	0	0	0	5	0	0	5
$babba$	10	0	0	0	0	10	0	0	0	0	0
$bbabb$	3	0	0	0	0	0	0	0	3	0	3
$babbab$	5	0	0	0	0	5	0	0	0	0	0

Relations :

$$aa = 0 \qquad aba = a \qquad b^3 = b^2 \qquad abbabb = 0 \qquad bbabba = 0$$

The idempotents are ab, ba, bb and 0. The $\mathcal{J}$-class structure is represented in the following diagram:

$$\begin{array}{|c|}\hline b \\ \hline\end{array}$$

$$\begin{array}{|c|c|}\hline {}^*ab & a \\ \hline bab & {}^*ba \\ \hline\end{array} \qquad \begin{array}{|c|}\hline {}^*b^2 \\ \hline\end{array}$$

$$\begin{array}{|c|}\hline abb \\ \hline babb \\ \hline\end{array} \qquad \begin{array}{|c|c|}\hline bba & bbab \\ \hline\end{array}$$

$$\begin{array}{|c|c|}\hline abba & abbab \\ \hline babba & babbab \\ \hline\end{array} \qquad \begin{array}{|c|}\hline bbabb \\ \hline\end{array}$$

$$\begin{array}{|c|}\hline {}^*0 \\ \hline\end{array}$$

The image of the language is $P = \{bbabb, abba, abbab, babba, babbab\}$. One can verify that P saturates $\equiv$. Notice in particular that $babbab = (ba)(bb)(ba)$. Since the elements $e = ba$ and $f = bb$ are idempotent, $efe \in P$ should imply $fef \in P$, since P saturates $\equiv$. Indeed, $fef = babba \in P$. In fact, $L = (F(ab^2, 1) \cap F(b^2a, 1)) \setminus (F(aa, 1) \cup F(ab^2, 2) \cup F(b^2a, 2))$.

5 Outline of the proof of Theorem 3.3

Our proof is partly inspirated by the proof of Wilke [27], which gives a very nice characterization of the TLT languages of infinite words. However, Wilke's characterization makes use of the topology on infinite words, which is useless on finite words. We first introduce some combinatorial definitions.

Let A be a finite alphabet. If u is a word of length $\geq k$, we denote by $p_k(u)$ and $s_k(u)$, respectively, the prefix and suffix of length k of u. If u and

x are two words, we denote by $\binom{u}{x}$ the number of occurrences of the factor x in u. For instance $\binom{abababa}{aba} = 3$, since aba occurs in three different places in $abababa$: $\underline{aba}baba$, $ab\underline{aba}ba$, $abab\underline{aba}$.

Let x and y be two integers. Then $x \equiv y$ threshold t (also denoted $x \equiv_t y$) if and only if ($x < t$ and $x = y$) or ($x \geq t$ and $y \geq t$). For instance the equivalence classes of $\equiv_4$ are $\{0\}$, $\{1\}$, $\{2\}$, $\{3\}$, $\{4, 5, 6, 7, \ldots\}$.

For every $k, t > 0$, let $\equiv_{k,t}$ be the equivalence of finite index defined on A^+ by setting $u \equiv_{k,t} v$ if and only if, for every word x of length $\leq k$, $\binom{u}{x} \equiv_t \binom{v}{x}$. For instance, $abababab \equiv_{2,3} abababa$ since $abababab$ contains 4 ($\equiv 3$ threshold 3) occurrences of ab and 3 ($\equiv 3$ threshold 3) occurrences of ba, and no occurrences of aa (respectively bb).

We also define a congruence $\sim_{k,t}$ of finite index on A^+ by setting $u \sim_{k,t} v$ if

(1) u and v have the same prefixes (resp. suffixes) of length $< k$,

(2) $u \equiv_{k,t} v$.

The next proposition gives an alternative definition of the TLT and STLT languages.

Proposition 5.1 *A subset of A^+ is TLT (resp. STLT) if it is union of $\sim_{k,t}$-classes (resp. $\equiv_{k,t}$) for some k and t.*

The equivalence of (1) and (2) follows from Theorem 3.1. We now prove that (2) implies (3). Let L be a STLT language. Then L is union of $\equiv_{k,t}$-classes for some k and t. Let $\eta : A^+ \to S$ be the syntactic morphism of L and let P be the syntactic image of L. Since L is STLT, it is also TLT and thus, by Theorem 2.2, S is aperiodic and satisfies (C). It remains to see that P saturates the $\equiv$-classes. Since η is onto, one can fix, for each element $s \in S^1$ a word $\bar{s} \in A^*$ such that $\eta(\bar{s}) = s$. Let s and r be two $\mathcal{J}$-equivalent elements of S and suppose that $s \in P$. Then there exist $x, y, u, v \in S^1$ such that $usx = r$ and $vry = s$.

Since S is finite, there is an integer n such that, for any $s \in S$, s^n is idempotent. Assuming that $n \geq kt$, one gets $(\bar{v}\bar{u})^n\bar{s}(\bar{x}\bar{y})^n \equiv_{k,t} \bar{u}(\bar{v}\bar{u})^n\bar{s}(\bar{x}\bar{y})^n\bar{x}$. But $\eta((\bar{v}\bar{u})^n\bar{s}(\bar{x}\bar{y})^n) = s \in P$ and thus $(\bar{v}\bar{u})^n\bar{s}(\bar{x}\bar{y})^n \in L$. It follows that $\bar{u}(\bar{v}\bar{u})^n(\bar{x}\bar{y})^n\bar{x} \in L$ and thus $\eta(\bar{u}(\bar{v}\bar{u})^n(\bar{x}\bar{y})^n\bar{x}) = r \in P$.

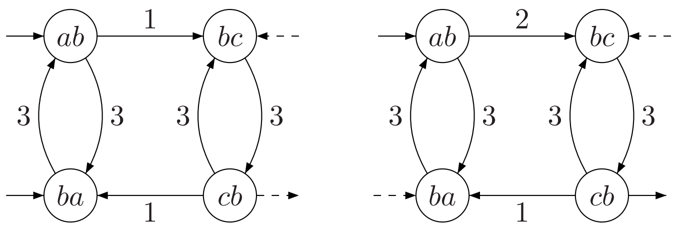
Let now e and f be two idempotents of S and suppose that $esfre \in P$. Then, for $n \geq kt$, $\bar{e}^n\bar{s}\bar{f}^n\bar{r}\bar{e}^n \equiv_{k,t} \bar{f}^n\bar{r}\bar{e}^n\bar{s}\bar{f}^n$. But $\eta(\bar{e}^n\bar{s}\bar{f}^n\bar{r}\bar{e}^n) = esfre \in P$ and thus $\bar{e}^n\bar{s}\bar{f}^n\bar{r}\bar{e}^n \in L$. Therefore $\bar{f}^n\bar{r}\bar{e}^n\bar{s}\bar{f}^n \in L$ and thus $\eta(f^nre^nsf^n) = fresf \in P$. Thus P saturates $\equiv$.

The direction (3) implies (2) is much more difficult. Since S is aperiodic and satisfies (C), Theorem 2.2 and Proposition 5.1 show that L is union of

$\sim_{k,t}$-classes for some k and t. Unfortunately, L is not in general union of $\equiv_{k,t}$-classes, but we will show that L is union of $\equiv_{k,T}$-classes for some large T (to be precise, one takes $T = (1 + t\cdot (|A|^k)!)(1 + |A|)$). Associate with each word u a labelled graph $N(u)$ defined as follows: the vertices are the words of length $k-1$, and if x is a word of length k, there is an edge of label $\binom{u}{x}$ threshold t from the prefix length $k-1$ of x to its suffix of length $k-1$. The prefix (resp. suffix) of u of length $k-1$ is called the initial (resp. final) vertex.

Thus let u and u' be two words such that $u \equiv_{k,T} u'$ and $u \in L$. Our aim is to show that $u' \in L$. If $|u| < T$ (or $|u'| < T$), then necessarily $u = u'$, thus we may assume that $|u|, |u'| \geq T$. Since $\binom{u}{x} = \binom{u'}{x}$ threshold T (and thus also threshold t), the labelled graphs $N(u)$ and $N(u')$ are equal, except for the initial and final vertices. We denote by i and f (resp. i' and f') the initial and final vertices of $N(u)$ (resp. $N(u')$).

In the figure below, two graphs are represented. The parameters are $k = 3$ and $t = 3$. The graph on the left hand side corresponds to the words $u = (ab)^4(cb)^4a$ and $u' = b(cb)^4(ab)^4cb$. The initial and final vertices of u (resp. u') are represented by full (resp. dotted) unlabelled arrows. The graph on the right hand side corresponds to the words $u = (ab)^4(cb)^4abcb$ and $u' = b(ab)^4(cb)^4acb$.



Two vertices v_1 and v_2 are in the same strongly t-component if there are two paths from v_1 to v_2 and from v_2 to v_1 using only edges of label t. For instance, in the two graphs above, ab and ba (resp. bc and cb) are in the same t-component. A non trivial combinatorial argument shows that if $u \equiv_{k,T} u'$, then two cases may arise:

(1) i and i' are in the same t-component and f and f' are in the same t-component,

(2) i and f are in the same t-component and i' and f' are in the same t-component.

The first and second cases are illustrated by the graphs on the right and on the left hand side, respectively. Now, one can show that in the first case, the elements $\eta(u)$ and $\eta(u')$ are $\mathcal{J}$-equivalent. Since P saturates the $\mathcal{J}$-classes, we are done in this case. In the second case, one can show that $\eta(u) \mathrel{\mathcal{J}} \eta(v)$ and $\eta(u') \mathrel{\mathcal{J}} \eta(v')$ for some words v and v' such that

(1) $v \equiv_{k,T} v'$,

(2) The initial and final vertices of $N(v)$ (resp. $N(v')$) coincide

But this last condition implies that for some idempotents e and f and for some elements p and q, $\eta(v) = epfqe$ and $\eta(v') = fqepf$. Now one can use the fact that P saturates the $\equiv$-classes.

References

[1] J. Almeida, *Finite semigroups and universal algebra*, Series in Algebra Vol 3, Word Scientific, Singapore, 1994.

[2] D. Beauquier and J.-E. Pin, Factors of words, in *Automata, Languages and Programming, (G. Ausiello, M. Dezani-Ciancaglini and S. Ronchi Della Rocca, eds.), Lecture Notes in Comput. Sci.* **372**, Springer, (1989), 63–79.

[3] D. Beauquier and J.-E. Pin, Languages and scanners, *Theoret. Comput. Sci.* **84**, (1991), 3–21.

[4] J.A. Brzozowski and I. Simon, Characterizations of locally testable languages, *Discrete Math.* **4**, (1973), 243-271.

[5] J.R. Büchi, On a decision method in restricted second-order arithmetic, in *Proc. 1960 Int. Congr. for Logic, Methodology and Philosophy of Science*, Stanford Univ. Press, Standford, (1962), 1–11.

[6] S. Cho and D.T. Huynh, Finite automaton aperiodicity is PSPACE-complete, *Theoret. Comput. Sci.* **88**, (1991), 99–116.

[7] Ebbinghaus and Flum, *Finite Model Theory*, Springer, (1995).

[8] S. Eilenberg, *Automata, languages and machines*, Vol. B, Academic Press, New York, 1976.

[9] H. Gaifman, On local and non-local properties, in *Proc. of the Herbrandt Symposium, Logic Colloquium'81 (J. Stern, ed.), Studies in Logic* **107**, North-Holland, Amsterdam, (1982), 105–135.

[10] R. McNaughton, Algebraic decision procedures for local testability, *Math. Syst. Theor.* **8**, (1974), 60-76.

[11] R. McNaughton and S. Pappert, *Counter-free Automata*, MIT Press, 1971.

[12] D. Perrin, *Automata*, Chapter 1 in Handbook of Theoretical Computer Science (Van Leeuwen, J. ed.), Vol B: Formal Models and Semantics, Elsevier (1990).

[13] D. Perrin and J.-E. Pin, First order logic and star-free sets, *J. Comput. System Sci.* **32**, (1986), 393–406.

[14] J.-E. Pin, *Variétés de langages formels*, Masson, Paris, 1984. English translation: *Varieties of formal languages*, Plenum, New-York, 1986.

[15] J.-E. Pin, Logic, Semigroups and Automata on Words, *Theor. Comp. Sc.*, to appear.

[16] J.-E. Pin, Finite semigroups and recognizable languages : an introduction, in NATO Advanced Study Institute *Semigroups, Formal Languages and Groups*, J. Fountain (ed.), Kluwer academic publishers, (1995), 1–32.

[17] J.-E. Pin, A variety theorem without complementation, *Izvestiya VUZ Matematika* **39** (1995) 80–90. English version, *Russian Mathem. (Iz. VUZ)* **39**, (1995), 74–83.

[18] M.P. Schützenberger, On finite monoids having only trivial subgroups, *Information and Control* **8**, (1965), 190–194.

[19] J. Stern, Characterization of some classes of regular events, *Theoret. Comp. Sci.* **35**, (1985), 17–42.

[20] J. Stern, Complexity of some problems from the theory of automata, *Inform. and Control* **66**, (1985), 63–176.

[21] H. Straubing, *Finite automata, formal logic and circuit complexity*, Birkhäuser, 1994.

[22] H. Straubing, D. Thérien and W. Thomas, Regular Languages Defined with Generalized Quantifiers, in *Proc. 15th ICALP*, Springer Lecture Notes in Computer Science **317**, (1988), 561–575.

[23] D. Thérien and A. Weiss, Graph congruences and wreath products, *J. Pure Applied Algebra* **35**, (1985), 205–215.

[24] W. Thomas, Classifying regular events in symbolic logic, *J. Comput. Syst. Sci* **25**, (1982), 360–375.

[25] W. Thomas, On logics, tilings, and automata, *Proc. 18th ICALP, Madrid, (J. Leach Albert et al., eds.), Lect. Notes in Comp. Sci.* **510**, Springer, Berlin, (1991), 441–454.

[26] W. Thomas, On the Ehrenfeucht-Fraïssé Game in Theoretical Computer Science, *TAPSOFT'93, M.C. Gaudel, J.P. Jouannaud (Eds.), Lect. Notes in Comp. Sci.* **668**, Springer, Berlin, (1993), 559–568.

[27] Th. Wilke, Locally threshold testable languages of infinite words, in *STACS 93, P. Enjalbert, A. Finkel, K.W. Wagner (Eds.), Lect. Notes in Comp. Sci.* **665**, Springer, Berlin, (1993), 607–616.